2023 IEEE International Conference on Development and Learning (ICDL)
November 9-11, Macau, China

Hierarchical Latent Dynamics Model with Multiple Timescales for Learning Long-Horizon Tasks

Kentaro Fujii[1] and Shingo Murata[1]

Abstract— **Long-horizon tasks require intelligent agents, such as robots, to handle both temporal uncertainty and temporal dependency. Although world models have shown promise for solving tasks across many domains, they often struggle with managing long context in tasks due to the limited representation ability of their latent dynamics models. To overcome this issue, we propose a novel hierarchical latent dynamics model that takes into account multiple-timescale dynamics. Specifically, our proposed model, called the "multiple timescale recurrent state-space model" (MTRSSM), comprises a higher level with slow dynamics and a lower level with fast dynamics, each incorporating both deterministic and stochastic latent states. We demonstrate, both quantitatively and qualitatively, that a world model with our proposed MTRSSM can generate superior video predictions for long-horizon robotic object-manipulation tasks through latent imagination compared with other baselines. Importantly, we emphasize the critical role of the higher level in effectively handling temporal uncertainty and temporal dependency in long-horizon tasks. These findings indicate that the proposed MTRSSM enables intelligent agents to acquire a better understanding of the environment and generate more accurate predictions, thereby facilitating their learning and planning of long-horizon tasks.**

I. Introduction

Performing long-horizon tasks is challenging for intelligent agents due to the presence of "temporal uncertainty" and "temporal dependency" in environmental and task dynamics. As an example, consider a robot tasked with making a pot of soup. To complete this task, the robot needs to perform multiple steps, including removing the lid of the pot, adding ingredients to the pot, closing the lid, moving the pot to a stove, and boiling the ingredients. Whereas the cooking steps must be followed in order, the sequence of ingredient addition may vary probabilistically. Moreover, although the scene looks the same before opening the lid as after closing it, the subsequent actions required depend on the context of cooking. This scenario reflects the challenge of temporal uncertainty and temporal dependency that makes long-horizon tasks difficult for intelligent agents to perform. To overcome these challenges, the robot must recognize the probability of the dynamics transition depending on its own actions and retain a memory of the completed steps.

Learning world models is a promising approach to predicting temporal uncertainty in environmental and task dynamics [1], [2]. For instance, the recurrent state-space model (RSSM) is a widely used latent dynamics model for world models [2], [3] because of its strong representation ability. However, RSSMs often have difficulty learning long-term temporal dependencies [4], [5]. One possible approach to solving this problem is introducing hierarchical levels with different time intervals [4], [6], [7]. Although this implementation of a temporal hierarchy can be effective in some scenarios, the use of a reset mechanism [6] and fixed intervals [4], [7] may not be suitable for continuously and dynamically changing real-world environments.

Another approach to learning temporal dependencies involves incorporating continuous multiple-timescale dynamics. The multiple timescale recurrent neural network (MTRNN) [8], inspired by the human brain's functional hierarchy, is known for its ability to learn long-term temporal dependencies. Nevertheless, its deterministic nature makes it difficult to represent temporal uncertainty. The predictive-coding-inspired RNN (PV-RNN) [9], which is a probabilistic extension of the MTRNN, is another candidate for learning both temporal uncertainty and temporal dependency. However, its backpropagation-based online inference process is computationally expensive, and thus applying it to real-world scenarios is not straightforward.

In this paper, we propose the "multiple timescale recurrent state-space model" (MTRSSM), which combines the advantages of the RSSM and MTRNN. The MTRSSM can learn not only temporal uncertainty through its stochastic latent representations but also long-term temporal dependencies through its hierarchical levels with continuous multiple-timescale dynamics. We evaluate the performance of our proposed model using complex real-world robotic data and show that it outperforms other baselines.

II. Related Work

A. World Model

World models, initially proposed by Ha and Schmidhuber [1], have shown great promise in the field of model-based reinforcement learning (RL). One of their key advantages is the ability to predict future latent states without the need for computationally expensive encoding and decoding observations, a concept known as "latent imagination." This advantage has led to the development of policies that generate actions based on the state of the world model, resulting in achievements in both RL [2], [3], [7], [10]–[13] and imitation learning (IL) [14], [15]. In addition, the RSSM has enhanced this advantage by incorporating both deterministic and stochastic latent states [2], [16], leading to achievements in simulated environments [2] as well as real-world environments [11]. However, the RSSM has difficulties in learning long-term temporal dependencies [4], [6], which

[1]Graduate School of Integrated Design Engineering, Keio University, Kanagawa, Japan oakwood.n14.4sp@keio.jp, murata@elec.keio.ac.jp

2023 IEEE International Conference on Development and Learning (ICDL) | 978-1-6654-7075-9/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICDL55364.2023.10364442

limits its performance on long-horizon tasks learned through latent imagination.

B. Hierarchy for Long-Horizon Imagination

Several models having a temporal hierarchy have been proposed for capturing temporal dependencies [4]–[7], [17]. In these models, higher levels with slower dynamics capture long-horizon transitions, while lower levels with faster dynamics focus on momentary changes in observations. For example, the variational temporal abstraction (VTA) [6] utilizes two RSSMs, with the lower level determining when to update the higher level. Similarly, the clockwork variational autoencoder (CW-VAE) [4] employs multiple RSSMs, with different clock speeds per level resulting in slower changes in the higher levels. However, the VTA exhibits discontinuous lower-level representations, and the clock speeds in the CW-VAE are fixed. Given the continuous and dynamic nature of real-world environments, the applicability of these models may be potentially limited.

In contrast, although our proposed MTRSSM also comprises two levels of RSSMs, it differs from the VTA and CW-VAE in its approach to implementing a temporal hierarchy. Instead of using a sparse update mechanism to introduce slow dynamics, we employ multiple-timescale dynamics, which we will elaborate upon in the following subsection.

C. Multiple Timescale Recurrent Neural Network

The MTRNN [8] is a hierarchically organized continuous-time RNN (CTRNN) in which each layer has a distinct time constant. Previous studies [8], [18] have demonstrated that the MTRNN can self-organize a functional hierarchy, where higher levels with slower dynamics encode combinations of primitive patterns represented by lower levels with faster dynamics. However, the MTRNN struggles to capture temporal uncertainty because of its deterministic hidden state dynamics. To overcome this limitation, the PV-RNN adds the capability of variational inference to the MTRNN, enabling the representation of temporal uncertainty in its latent dynamics. However, because the PV-RNN is a model of dynamic predictive coding, it requires gradient-based optimization with backpropagation through time (BPTT) to realize variational inference. This limitation prevents its practical application in real-world environments, including action generation by robots.

In contrast, our proposed MTRSSM combines the deterministic state dynamics of the MTRNN with the stochastic state dynamics of the RSSM. Additionally, we leverage amortized inference, similar to the RSSM, to realize variational inference, thereby eliminating the need for gradient-based optimization with BPTT. This leads to improved time efficiency compared with the PV-RNN, making our approach more suitable for practical applications in real-world environments.

III. Methods

A. World Model

The world model in this study comprises our proposed MTRSSM, which is used as a latent dynamics model, as well as an image encoder and an image decoder (Fig. 1). The formulation of the MTRSSM is described below.

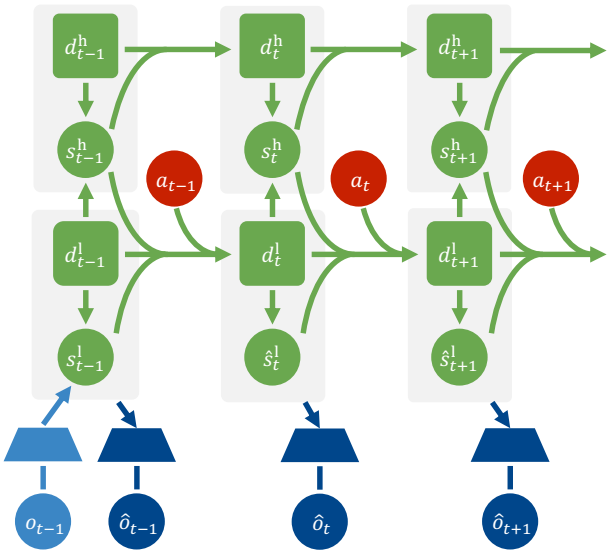


Fig. 1. The world model employed in this study. The model comprises our proposed MTRSSM, which is used as a latent dynamics model, as well as an image encoder and an image decoder. The higher level of the MTRSSM changes slowly while the lower level changes quickly. Generating video predictions without external inputs can be realized by "latent imagination" using the priors of the MTRSSM, as shown in the figure after time step t. The latent state z_t is divided into the lower-level latent state z^{l}_t and the higher-level latent state z^{h}_t, where the latent state in each level is the concatenation of the deterministic state d_t and the stochastic state s_t.

The MTRSSM comprises a lower-level RSSM with faster dynamics and a higher-level RSSM with slower dynamics. Each level has its own latent states, which are concatenations of a deterministic state and a stochastic state. Specifically, at time step $t \in [1, T]$, the latent state z_t is divided into the lower-level latent state z^{l}_t and the higher-level latent state z^{h}_t, where the latent state in each level is the concatenation of the deterministic state d_t and the stochastic state s_t. Using this architecture, we expect that the lower level learns to represent primitive patterns and their uncertainty, while the higher level learns to capture combinations of the primitive patterns and their uncertainty.

The deterministic state d_t is described by the following deterministic function f_θ based on the hidden state dynamics of the MTRNN:

$$\begin{aligned} u_t &= \left(1 - \frac{1}{\tau}\right) u_{t-1} + \frac{1}{\tau}\left(W_{\mathrm{xd}} x_t + W_{\mathrm{dd}} d_{t-1} + b_{\mathrm{d}}\right) \\ d_t &= \tanh(u_t), \end{aligned} \tag{1}$$

where u_t is the internal state before activation, x_t is the input state, and τ is the time constant. A larger value of τ corresponds to slower changes in the state. In this study, we set the time constant for the lower level as $\tau^{\mathrm{l}} = 8$, and that for the higher level as $\tau^{\mathrm{h}} = 32$.

The stochastic state s_t is represented by a set of one-hot vectors sampled from the prior or posterior with categorical distribution. The lower-level and higher-level priors, p^{l}_θ and p^{h}_θ, are predicted from the deterministic state in the same

level, d_t^{l} and d_t^{h}, respectively. In contrast, the lower-level posterior q_θ^{l} is inferred from the observation o_t and the deterministic state d_t^{l} in the same level, while the higher-level posterior q_θ^{h} is inferred from both the lower-level and higher-level deterministic states, d_t^{l} and d_t^{h}, respectively.

The image encoder uses a convolutional neural network (CNN) to embed a high-dimensional visual observation o_t into the low-dimensional feature state. The decoder uses a transposed CNN to reconstruct the visual observation as $\hat{o}_t$ from the lower-level latent state z_t^{l}.

In summary, the MTRSSM components are:

$$\begin{aligned}
&\text{Model state:} && z_t = \left(z_t^{\mathrm{l}}, z_t^{\mathrm{h}}\right)\\
& && z_t^{\mathrm{l}} = \left(d_t^{\mathrm{l}}, s_t^{\mathrm{l}}\right)\\
& && z_t^{\mathrm{h}} = \left(d_t^{\mathrm{h}}, s_t^{\mathrm{h}}\right)\\
&\text{Lower Layer}\\
&\quad\text{Deterministic state:} && d_t^{\mathrm{l}} = f_\theta^{\mathrm{l}}\left(z_{t-1}^{\mathrm{l}}, s_{t-1}^{\mathrm{h}}, a_{t-1}\right)\\
&\quad\text{Prior:} && \hat{s}_t^{\mathrm{l}} \sim p_\theta^{\mathrm{l}}\left(\hat{s}_t^{\mathrm{l}} \mid d_t^{\mathrm{l}}\right)\\
&\quad\text{Approximate Posterior:} && s_t^{\mathrm{l}} \sim q_\theta^{\mathrm{l}}\left(s_t^{\mathrm{l}} \mid d_t^{\mathrm{l}}, o_t\right)\\
&\text{Higher Layer}\\
&\quad\text{Deterministic state:} && d_t^{\mathrm{h}} = f_\theta^{\mathrm{h}}\left(\hat{z}_{t-1}^{\mathrm{h}}\right)\\
&\quad\text{Prior:} && \hat{s}_t^{\mathrm{h}} \sim p_\theta^{\mathrm{h}}\left(\hat{s}_t^{\mathrm{h}} \mid d_t^{\mathrm{h}}\right)\\
&\quad\text{Approximate Posterior:} && s_t^{\mathrm{h}} \sim q_\theta^{\mathrm{h}}\left(s_t^{\mathrm{h}} \mid d_t^{\mathrm{l}}, d_t^{\mathrm{h}}\right).
\end{aligned} \tag{2}$$

Note that the higher level does not have any observation or action inputs. The only input from the lower level to the higher level is the deterministic state for inferring the approximate posterior.

All model components are jointly trained in order to minimize the following negative value of the modified evidence lower bound objective (ELBO), similar to the VAE [19], RSSM [2], [3], and CW-VAE [4]:

$$\begin{aligned}
\mathcal{L}(\theta) &= \sum_{t=1}^{T} \mathrm{E}_{q_\theta^{\mathrm{l}}(z_t^{\mathrm{l}} \mid a_{t-1}, o_t) q_\theta^{\mathrm{h}}(z_t^{\mathrm{h}})}\left[\mathcal{L}_{\mathrm{recon}}(\theta) + \mathcal{L}_{\mathrm{KL}}(\theta)\right],\\
\mathcal{L}_{\mathrm{recon}}(\theta) &= -\log p_\theta\left(o_t \mid z_t^{\mathrm{l}}\right) = \left[\left(\hat{o}_t - o_t\right)^2 + \epsilon^2\right]^{\frac{1}{2}},\\
\mathcal{L}_{\mathrm{KL}}(\theta) &= \beta^{\mathrm{l}} D_{\mathrm{KL}}\left[q_\theta^{\mathrm{l}}\left(s_t^{\mathrm{l}} \mid d_t^{\mathrm{l}}, o_t\right) \,||\, p_\theta^{\mathrm{l}}\left(\hat{s}_t^{\mathrm{l}} \mid d_t^{\mathrm{l}}\right)\right]\\
&\quad + \beta^{\mathrm{h}} D_{\mathrm{KL}}\left[q_\theta^{\mathrm{h}}\left(s_t^{\mathrm{h}} \mid d_t^{\mathrm{l}}, d_t^{\mathrm{h}}\right) \,||\, p_\theta^{\mathrm{h}}\left(\hat{s}_t^{\mathrm{h}} \mid d_t^{\mathrm{h}}\right)\right],
\end{aligned} \tag{3}$$

where D_{KL} denotes the Kullback–Leibler (KL) divergence, and β^{l} and β^{h} are the weights of the KL divergence terms. For the reconstruction loss $\mathcal{L}_{\mathrm{recon}}(\theta)$, we assume the Charbonnier loss [20], [21]. In this study, we set $\epsilon = 10^{-3}$, $\beta^{\mathrm{l}} = 1$, and $\beta^{\mathrm{h}} = 0.1$.

B. Behavior Cloning Policy

The policy employed in this study is a simple behavior cloning (BC) policy that learns to mimic expert demonstrations. This BC policy generates actions from the latent states of the world model, and the generated actions are utilized for the latent imagination. To reduce overlaps [8], different action patterns are sparsely encoded by categorical distributions through softmax transformation [22], [23]. The gripper state (opening or closing) is represented by binary encoding. We use a single-layer CTRNN to generate smooth actions. For the time constant, we set a relatively smaller value $\tau^{\mathrm{BC}} = 4$ compared with that of the RSSMs in order to prevent inheriting past information in the policy. The policy is described as follows:

$$\begin{aligned}
\left(h_t^{\text{arm}}, h_t^{\text{gripper}}\right) &= \mathrm{MLP}_\psi\left(\mathrm{CTRNN}_\psi\left(z_t^{\mathrm{h}}, z_t^{\mathrm{l}}\right)\right)\\
\hat{a}_t^{\text{arm}} &= \mathrm{softmax}\left(h_t^{\text{arm}}\right)\\
\hat{a}_t^{\text{gripper}} &= \mathrm{sigmoid}\left(h_t^{\text{gripper}}\right).
\end{aligned} \tag{4}$$

In practice, the policy predicts the action five steps ahead a_{t+5} from the current latent state z_t. The predicted actions are appended to a buffer and utilized as the action input after five time steps.

The BC policy is trained to minimize the following cross entropy loss:

$$\begin{aligned}
\mathcal{L}(\psi) &= \beta^{\text{arm}}\mathcal{L}^{\text{arm}}(\psi) + \beta^{\text{gripper}}\mathcal{L}^{\text{gripper}}(\psi)\\
\mathcal{L}^{\text{arm}}(\psi) &= \sum_{t=1}^{T}\sum_{\text{arm}} -\tilde{a}_t^{\text{arm}} \log\left(\hat{a}_t^{\text{arm}}\right)\\
\mathcal{L}^{\text{gripper}}(\psi) &= \sum_{t=1}^{T}\sum_{\text{gripper}} -\tilde{a}_t^{\text{gripper}} \log\left(\hat{a}_t^{\text{gripper}}\right),
\end{aligned} \tag{5}$$

where $\tilde{a}_t$ denotes expert demonstrations, and β^{arm} and β^{gripper} are the weights of the arm and gripper losses, respectively. In this study, we set $\beta_{\text{arm}} = 1$ and $\beta_{\text{gripper}} = 10$.

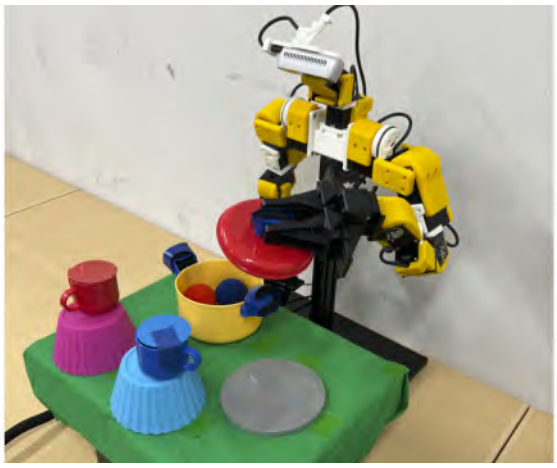


Fig. 2. Experimental environment. A dual-arm robot named Rakuda-2 performed long-horizon tasks involving the manipulation of multiple objects.

IV. Experimental Setup

A. Task Setting

To evaluate our proposed model, we designed a long-horizon robotic object-manipulation task that emulates a cooking procedure. In the task space shown in Fig. 2, a dual-arm robot named Rakuda-2, which was developed by ROBOTIS Japan, faces a table on which there are various objects, including a pot, its lid, a stove, red and blue balls, cups, and upturned bowls.

978-1-6654-7075-9/23/$31.00 ©2023 IEEE

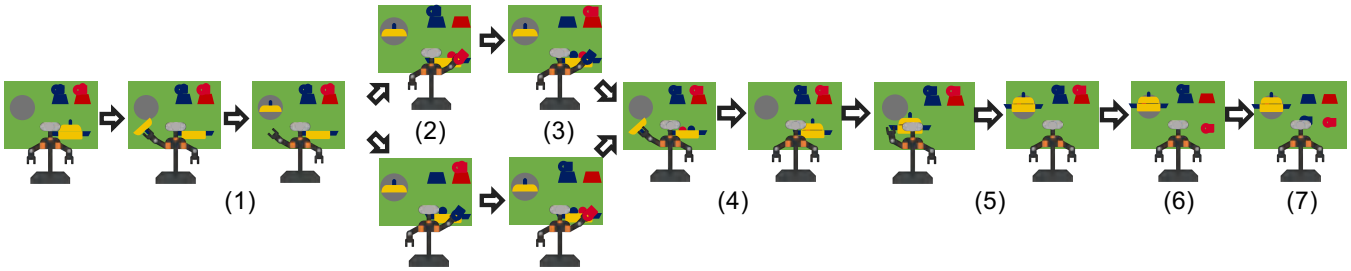


Fig. 3. Flow of a long-horizon robotic object-manipulation task. The robot sequentially manipulates multiple objects. The task includes a probabilistic branch after step (1), which represents temporal uncertainty. Additionally, the visual scene after step (4) is identical to the initial scene because of the temporal dependency of the task.

In this task, the robot was required to manipulate the objects from an initial fixed state to a final fixed state while considering both temporal uncertainty and temporal dependency. Specifically, the robot was required to (1) pick up the pot lid and place it on the stove, (2) pick up the red (or blue) cup and put the red (or blue) ball inside the cup in the pot, (3) pick up the blue (or red) cup and put the blue (or red) ball inside the cup in the pot, (4) pick up the pot lid and place it on the pot, (5) pick up the pot and place it on the stove, (6) pick up the red cup and place it on the front of the table, and (7) pick up the blue cup and place it on the front of the table (Fig. 3). After the first step, the robot was allowed to probabilistically decide the order for manipulating the red and blue cups. This required the robot to recognize the probabilitiy of the dynamics transition depending on the action. Moreover, the visual scene after the fourth step was identical to the initial scene. This required the robot to retain a memory of the completed steps in order to decide the subsequent actions. We consider that the world model with our proposed MTRSSM efficiently captures and represents both the temporal uncertainty and temporal dependency.

For the training of the world model and BC policy, we collected demonstration data by controlling the robot in a leader–follower manner. Specifically, 6-degree-of-freedom (DoF) arms, 1-DoF grippers, and a 1-DoF torso (yaw direction) were controlled. To fix the view from the camera mounted on the head, the head yaw direction was automatically controlled to face the opposite direction of the torso yaw direction.

B. Training Details

We collected 50 sets of demonstration data containing the total time-series data of 15-dimensional joint angles as actions and $64 \times 48 \times 3$-dimensional camera images as visual observations. Half of the sets are for the upper flow in Fig. 3, and the remaining sets are for the lower flow. One set of demonstration data from each flow was used for validation, and the rest were used for training. Each set of demonstration data contains $T = 1,000$ time steps, and the entire steps were used to train the world model and BC policy. The learning rate for both the world model and BC policy was set to 0.01, and AdaBerief [24] was used for parameter optimization. The world model and BC policy were trained for 4,000 and 10,000 epochs, respectively. For testing, we used the world model and BC policy at the epoch with the lowest validation loss throughout the epochs.

C. Baselines

We compared our proposed MTRSSM with three other models: the original RSSM, a hierarchical RSSM (H-RSSM), and a clockwork RSSM (CW-RSSM). The architecture of the original RSSM is the same as that used in the DreamerV2 [3]. The architecture of the H-RSSM is the same as that of the MTRSSM, except that it employs gated recurrent units (GRUs) [25] for the deterministic function f_θ at each level, instead of Eq. 1. The architecture of the CW-RSSM is similar to that of the H-RSSM, except that the higher level of the CW-RSSM is updated at a slower, fixed interval compared with the lower level. Specifically, the higher level is updated every eight steps. This update mechanism is derived from the CW-VAE [4]. The architecture of the encoder and decoder of the world model as well as the BC policy and the number of stochastic states are shared across all the models. The number of deterministic states (MTRNN or GRU) is adjusted so that all models have a nearly equal number of parameters for a fair comparison.

D. Video Prediction Using Latent Imagination

To compare the representation capability of the proposed model and baselines, we generated video predictions using latent imagination. During latent imagination, visual observations and actions from a demonstration were provided to each model for the initial 10 steps. Using the last state inferred from the approximate posterior, the subsequent state and action were predicted from the prior and policy, respectively. Predicted states and actions were then utilized to predict subsequent states and actions. In addition, by providing the predicted states to the image decoder, we were able to obtain video predictions without any external inputs after the initial 10 steps. We used the two hold-out (validation) data for the initial 10 steps and generated five video predictions from each by sampling stochastic states.

V. Results and Discussion

A. Video Prediction

We computed the structural similarity index (SSIM; higher is better), peak signal-to-noise ratio (PSNR; higher is better), and learned perceptual image patch similarity (LPIPS; lower

978-1-6654-7075-9/23/$31.00 ©2023 IEEE

TABLE I
QUANTITATIVE EVALUATION OF VIDEO PREDICTIONS BY MTRSSM AND BASELINES.

Model	SSIM	PSNR	LPIPS
MTRSSM (ours)	**0.458**	**14.321**	**0.138**
RSSM	0.371	13.119	0.182
H-RSSM	0.334	12.058	0.205
CW-RSSM	0.403	13.509	0.165

is better) [26] for each of the 10 video predictions and each of the 50 demonstrations. These metrics were then averaged across the 500 combinations and 1,000 time steps. The results are summarized in Table I. Our proposed MTRSSM outperformed all of the baselines in all evaluation metrics. At the same time, the H-RSSM, a hierarchical extension of the RSSM, performed worse than the RSSM on all evaluation metrics. This suggests that the simple introduction of a hierarchical structure did not enable the higher level to learn useful representations for long-term video predictions. In contrast, the CW-RSSM outperformed the RSSM, indicating that the slow update mechanism of the higher level enabled the model to retain the long context. The crucial difference between the proposed MTRSSM and the CW-RSSM is their update mechanism. Specifically, the higher level of the MTRSSM changes slowly in a continuous manner depending on its time constant, while that of the CW-RSSM changes slowly in a more discrete manner depending on its update interval. This difference in the higher level dynamics might have affected the representation ability of each model.

We visualized the video predictions of the MTRSSM with the best and worst LPIPS, as well as those of the baselines with the worst LPIPS in Fig. 4 along with the ground truth. Although the best predictions of the MTRSSM captured detailed parts such as small balls, the worst predictions of the MTRSSM failed to accurately represent these parts, and some images are distorted. However, regardless of the best and worst predictions, the MTRSSM succeeded in representing the overall flow of the long-horizon task. The RSSM produced similar images throughout all time steps, which can be attributed to its difficulty in representing long-term temporal dependency. Similarly, the H-RSSM produced similar images until around time step 200, after which it started producing distorted images after time step 300. The CW-RSSM successfully maintained the task flow until around time step 600 but then produced the pot lid in unstable positions thereafter.

These quantitative and qualitative results indicate that the introduction of hierarchical levels with continuous multiple-timescale dynamics enables the learning of long-term temporal dependency.

B. Analysis of Latent State Representation

To analyze how temporal uncertainty and temporal dependency are represented in the MTRSSM, we visualized the stochastic state dynamics of the higher level and the deterministic state dynamics of both the higher and lower levels as well as the deterministic state dynamics of the RSSM for comparison. Note that for this analysis, we used the states inferred from the approximate posterior by providing visual observations and actions from demonstrations. The results are shown in Fig. 5. In the figure, a subtle yet noteworthy representation can be observed, particularly in the first row with ("dim 0"). Around time step 200 (indicated by the red arrows in the figure), two classes ("0" and "2") are gray, indicating that one of them can be sampled. This time step corresponds to the second step of the task flow, where the robot probabilistically decides the order for manipulating the red and blue cups. Further analysis revealed that the sampled stochastic latent states at this point significantly influenced the decision regarding the manipulation order in the second step. Thus, the higher-level stochastic latent states serve as a representation of temporal uncertainty in the long-horizon task.

The deterministic state dynamics of the MTRSSM and RSSM were analyzed using principal component analysis (PCA), as illustrated in Fig. 6. The higher-level deterministic states of the MTRSSM in Fig. 6a exhibit a branching pattern around time step 200 and converge between time steps 400 and 500. These time steps correspond to the branch point in the task structure depicted in Fig. 3. Moreover, despite having the same visual observations, the states at the initial time step and at time step 700 are significantly different. Meanwhile, the lower-level deterministic states of the MTRSSM in Fig. 6b display a similar pattern at these time steps. Similarly, the deterministic states of the RSSM in Fig. 6c also demonstrate a similar pattern at these steps. These findings indicate that the deterministic states of the lower level in the MTRSSM and those of the RSSM are unable to differentiate the same visual observations with different contexts. Consequently, we can conclude that the higher-level deterministic states of the MTRSSM plays a crucial role in maintaining long-term temporal dependency.

VI. CONCLUSIONS

In this study, we presented the MTRSSM, a hierarchical extension of the conventional RSSM, aimed at learning long-horizon robotic tasks with temporal uncertainty and temporal dependency. The proposed model is characterized by its hierarchical latent representations, consisting of deterministic states with continuous multiple-timescale dynamics and stochastic states. The experimental results—specifically the video predictions generated by the latent imagination—demonstrated that our proposed MTRSSM outperformed the other baselines. These findings indicate the superiority of our approach in capturing and predicting complex long-term task dynamics. Furthermore, analysis of the latent representations of the MTRSSM revealed that temporal uncertainty and temporal dependency in long-horizon tasks are represented by higher-level stochastic and deterministic states, respectively. In future work, we plan to evaluate the capability of our proposed MTRSSM for real-world robot control.

978-1-6654-7075-9/23/$31.00 ©2023 IEEE

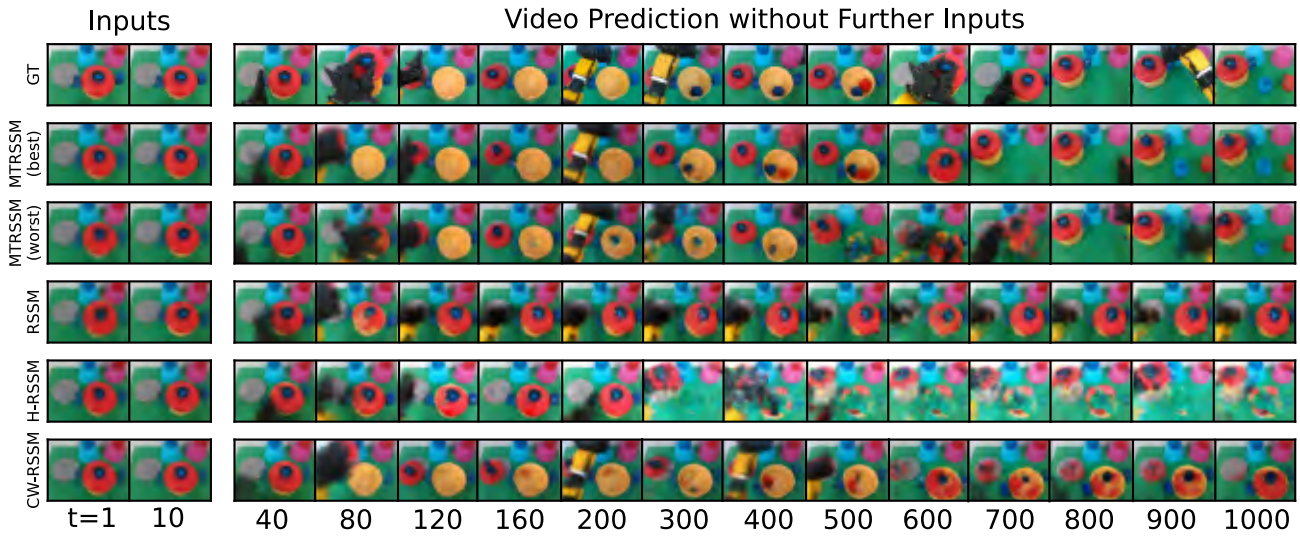


Fig. 4. Video predictions through latent imagination by our proposed MTRSSM and the baselines. The upper row shows an example of ground truth images collected during human demonstrations. From the subsequent row, the video predictions of the MTRSSM with the best and worst LPIPS, and those of the RSSM, H-RSSM, and CW-RSSM with the worst LPIPS are shown. To generate these video predictions, visual observations and actions from a demonstration were provided to each model for the initial 10 steps and images for the subsequent steps were generated through latent imagination without any external inputs.

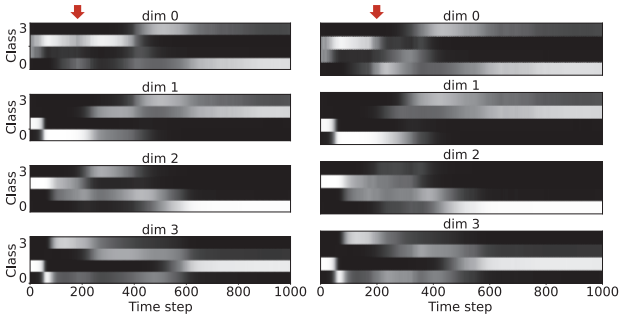


Fig. 5. Higher-level stochastic latent state dynamics corresponding to the upper flow (left) and the lower flow (right) in Fig. 3. At each time step, in each dimension, one of the four classes was randomly selected based on the distribution represented by grayscale, where white represents higher probability and black represents lower probability. In "dim 0," around time step 200 (indicated by the red arrows) two classes ("0" and "2") are gray, indicating that one of them can be probabilistically sampled at this moment.

ACKNOWLEDGMENT

This work was supported by the Japan Science and Technology Agency (PRESTO Grant No. JPMJPR22C9).

REFERENCES

[1] D. Ha and J. Schmidhuber, "Recurrent world models facilitate policy evolution," *Advances in neural information processing systems*, vol. 31, pp. 2450–2462, 2018.

[2] D. Hafner, J. Pasukonis, J. Ba, and T. Lillicrap, "Mastering diverse domains through world models," *arXiv preprint arXiv:2301.04104*, 2023.

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

[4] V. Saxena, J. Ba, and D. Hafner, "Clockwork variational autoencoders," in *Advances in Neural Information Processing Systems*, M. Ranzato, A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, Eds., vol. 34. Curran Associates, Inc., 2021, pp. 29 246–29 257.

[5] W. Cai, T. Wang, J. Wang, and C. Sun, "Learning a world model with multitimescale memory augmentation," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–10, 2022.

[6] T. Kim, S. Ahn, and Y. Bengio, "Variational temporal abstraction," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[7] D. Hafner, K.-H. Lee, I. Fischer, and P. Abbeel, "Deep hierarchical planning from pixels," in *Advances in Neural Information Processing Systems*, 2022.

[8] Y. Yamashita and J. Tani, "Emergence of functional hierarchy in a multiple timescale neural network model: a humanoid robot experiment," *PLoS computational biology*, vol. 4, no. 11, p. e1000220, 2008.

[9] A. Ahmadi and J. Tani, "A novel predictive-coding-inspired variational rnn model for online prediction and recognition," *Neural computation*, vol. 31, no. 11, pp. 2025–2074, 2019.

[10] D. Hafner, T. Lillicrap, J. Ba, and M. Norouzi, "Dream to control: Learning behaviors by latent imagination," in *International Conference on Learning Representations*, 2020.

[11] P. Wu, A. Escontrela, D. Hafner, P. Abbeel, and K. Goldberg, "Daydreamer: World models for physical robot learning," in *Conference on Robot Learning*. PMLR, 2023, pp. 2226–2240.

[12] M. Okada and T. Taniguchi, "Dreaming: Model-based reinforcement learning by latent imagination without reconstruction," *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 4209–4215, 2021.

[13] ——, "Dreamingv2: Reinforcement learning with discrete world models without reconstruction," *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 985–991, 2022.

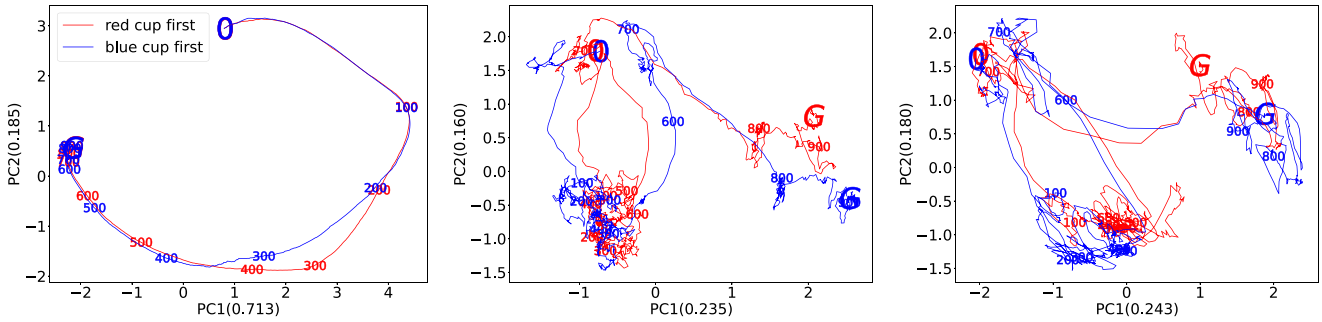
[14] R. Wei, A. Garcia, A. McDonald, G. Markkula, J. Engström, I. Supeene, and M. O'Kelly, "World model learning from demonstrations with active inference: application to driving behavior," in *International Workshop on Active Inference*. Springer, 2022, pp. 130–142.

[15] B. DeMoss, P. Duckworth, N. Hawes, and I. Posner, "Ditto: Offline imitation learning with world models," *arXiv preprint arXiv:2302.03086*, 2023.

[16] D. Hafner, T. Lillicrap, I. Fischer, R. Villegas, D. Ha, H. Lee, and J. Davidson, "Learning latent dynamics for planning from pixels," in *International conference on machine learning*. PMLR, 2019, pp. 2555–2565.

[17] J. Chung, S. Ahn, and Y. Bengio, "Hierarchical multiscale recurrent neural networks," *arXiv preprint arXiv:1609.01704*, 2016.

[18] S. Murata, Y. Li, H. Arie, T. Ogata, and S. Sugano, "Learning to achieve different levels of adaptability for human–robot collaboration



(a) (b) (c)

Fig. 6. Deterministic state dynamics of the MTRSSM and RSSM. (a) Higher-level deterministic state dynamics of the MTRSSM. (b) Lower-level deterministic state dynamics of the MTRSSM. (c) Deterministic state dynamics of the RSSM. Original dimensions are reduced to two by principal component analysis. The labels "0" and "G" represent the initial state and goal (last) state, respectively, and the numbers such as "0", "100", and "200" indicate the time steps of the states. The red and blue trajectories correspond to the states for the upper and lower flow, respectively, in Fig. 3.

utilizing a neuro-dynamical system," *IEEE Transactions on Cognitive and Developmental Systems*, vol. 10, no. 3, pp. 712–725, 2018.

[19] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," in *International Conference on Learning Representations*, 2014.

[20] J. T. Barron, "A general and adaptive robust loss function," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4331–4339.

[21] P. Charbonnier, L. Blanc-Feraud, G. Aubert, and M. Barlaud, "Two deterministic half-quadratic regular- ization algorithms for computed imaging," in *1st International Conference on Image Processing*, 1994.

[22] C. M. Bishop and N. M. Nasrabadi, *Pattern recognition and machine learning*. Springer, 2006, vol. 4, no. 4.

[23] A. Ahmadi and J. Tani, "How can a recurrent neurodynamic predictive coding model cope with fluctuation in temporal patterns? robotic experiments on imitative interaction," *Neural Networks*, vol. 92, pp. 3–16, 2017.

[24] J. Zhuang, T. Tang, Y. Ding, S. C. Tatikonda, N. Dvornek, X. Papademetris, and J. Duncan, "Adabelief optimizer: Adapting stepsizes by the belief in observed gradients," *Advances in neural information processing systems*, vol. 33, pp. 18 795–18 806, 2020.

[25] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," in *Advances in Neural Information Processing Systems*, 2014.

[26] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 586–595.