PreLAR: World Model Pre-training with Learnable Action Representation

Lixuan Zhang[1,2], Meina Kan[1,2], Shiguang Shan[1,2,3], and Xilin Chen[1,2]

[1] Key Laboratory of AI Safety of CAS, Institute of Computing Technology, Chinese Academy of Sciences (CAS), Beijing 100190, China
[2] University of Chinese Academy of Sciences, Beijing 100049, China
[3] Peng Cheng Laboratory, Shenzhen, 518055, China
lixuan.zhang@vipl.ict.ac.cn, {kanmeina, sgshan, xlchen}@ict.ac.cn

Abstract. The recent technique of Model-Based Reinforcement Learning learns to make decisions by building a world model about the dynamics of the environment. The world model learning requires extensive interactions with the real environment. Therefore, several innovative approaches such as APV proposed to unsupervised pre-train the world model from large-scale videos, allowing fewer interactions to fine-tune the world model. However, these methods only pre-train the world model as a video predictive model without action conditions, while the final world model is action-conditional. This gap limits the effectiveness of unsupervised pre-training in enhancing the world model's capabilities. To further release the potential of unsupervised pre-training, we introduce an approach that Pre-trains the world model from action-free videos but with Learnable Action Representation (PreLAR). Specifically, the observations of two adjacent time steps are encoded as an implicit action representation, with which the world model is pre-trained as action conditional. To make the implicit action representation closer to the real action, an action-state consistency loss is designed to self-supervise its optimization. During fine-tuning, the real actions are encoded as the action representation to train the overall world model for downstream tasks. The proposed method is evaluated on various visual control tasks from the Meta-world simulation environment. The results show that the proposed PreLAR significantly improves the sample efficiency in world model learning, demonstrating the necessity of incorporating action in the world model pre-training. Codes can be found at https://github.com/zhanglixuan0720/PreLAR

Keywords: World Model · Unsupervised Learning · Model-Based Reinforcement Learning · Learnable Action Representation

1 Introduction

The technique of Model-Based Reinforcement Learning learns to make decisions by building a dynamics model of the environment [26]. It follows the framework of an agent interacting in an environment, learning a model of the environment,

and then leveraging the model for decision-making. Equipped with a learned dynamics model of the environment, a system uses this model of the world to ask questions of the form "what will happen if I do x?" to choose the best x [3]. In this manner, the agent is enabled to interact with this model rather than the actual environment to make decisions or learn behaviors more efficiently.

The recent advanced works propose to build a world model that further develops the dynamics model by not only approximating the state transition but also reward prediction of the actual environment to facilitate planning and behavior learning (akin the System 2 and System 1 in [7] respectively) with imaginary trajectories [22]. In previous work, the world model has been studied and achieved tremendous success by learning latent dynamics models in compact representation space [14–17] or action-conditional video prediction models [10], such as the DreamerV2 [16] and VLP [10]. However, learning the world model from scratch without any preliminary knowledge of the environment demands substantive interactions with the real environment, as the learning of the dynamics needs feedback from the environment. This substantive interaction usually takes a lot of time or is even impossible in some scenarios, which thus limits the efficiency of the word model and behavior learning. Inspired by the success of unsupervised pre-training in fields like computer vision [9] and natural language processing [27], some researchers have suggested unsupervised training of world models using unlabeled data, such as the videos without action and reward labels. This pre-training methodology serves to infuse the world model with prior knowledge derived from the unlabeled videos. Consequently, only minimal interactions are needed to fine-tune the world model for a specific downstream task, thereby significantly enhancing learning efficiency.

In the context of the world model, the pre-training and fine-tuning paradigm was first investigated by recent innovative work, Action-free Pre-training from Videos (APV) [30]. The APV pretrains a video predictive model from action-free videos to learn the representation useful for understanding the dynamics. In the fine-tuning phase, an action-conditional latent dynamics model is stacked on top of the video predictive model and fine-tuned with actual interaction data from the environment. In this way, the learned state representations transfer well to downstream tasks. Afterward, the ContextWM [33] world model poses an improvement of the APV, which models visual context and dynamics individually to remove the interference of complex and diverse visual information irrelevant to the dynamics transition, improving the learning efficiency of the world model.

However, these two methods only pre-train the world model as a video predictive model without considering action conditions, whereas the final world model of a specific task is action conditional. The video predictive model is largely engaged in capturing the transition *dependencies* that exist between sequential states, while the world model concentrates on learning the *causality* that connects these sequential states. This discrepancy causes a modeling gap between the pre-training and the fine-tuning phases. Consider a scenario where a robot arrives at an intersection with multiple possible routes like turning left or right. The video predictive model is only equipped to capture the probability of the

robot turning either way based on the dataset. On the contrary, the world model is expected to learn the causality between the turning action and the result views, such as `'turn left'` action leads to a view of left, while `'turn right'` action leads to a view of right. Therefore, pre-training the world model as a video predictive model falls short in modeling the causal transition which may restrict full potential of unsupervised pre-training from action-free videos.

To further release the potential of unsupervised pre-training in the world model, in this work, we propose to pre-train the world model from action-free videos but with learnable action representation, i.e. pre-trains the world model as action-conditional. Specifically, we achieve this by encoding the observations of two adjacent time steps into an implicit action representation and thereby facilitating the pre-training of the world model as action-conditional. Furthermore, to make this implicit action representation closer to the real action, we design an action-state consistency loss to self-supervise its optimization. During the fine-tuning phase, real actions are encoded as the action representation to train the overall world model for specific downstream tasks. Our proposed action-conditional pre-training allows the implicit learning of transition causality in the videos, instead of just capturing the transition dependencies in the action-free pre-training. This thus reduces the discrepancy between the pre-training and fine-tuning, inducing a better transfer of the pre-training model to specific downstream tasks. Briefly, the main contributions of this work are as follows:

- We propose an action-conditional pre-training scheme for the world model, which pre-trains the world model from videos with Learnable Action Representation (PreLAR). This kind of pre-training implicitly learns of transition causality from action-free videos and thus facilitates the fine-tuning of the world model for specific downstream tasks.
- To better learn the implicit action representation, we design an inverse dynamics encoder as well as an action-state consistency loss to self-supervise the optimization of implicit action representation.
- Superior performance is achieved in various visual control tasks from the Meta-world simulation environment, which demonstrates the effectiveness of our method in improving the sample efficiency of world model learning.

2 Related Work

In this section, we first introduce the development and architecture of the world model in reinforcement learning. Afterward, several representative approaches in world model pre-training are discussed. Finally, we investigate the action representation in robotics when learning behavior from action-less videos.

2.1 World Model in Reinforcement Learning

Model-Based Reinforcement Learning (MBRL) enhances decision-making by building a dynamics model of the environment, thereby reducing significant interaction with the real world during the trial-and-error process and improving

learning efficiency. Recent advancements in the concept of the world model augment the dynamics model by approximating the actual environment's state transition and reward prediction, thereby facilitating planning and behavior learning with imaginary trajectories [22]. Ha and Schmidhuber first propose this concept, defining the world model as a latent dynamics model within a compact representation space, with the latent state representation compressed from visual observation through a Variational Auto-Encoder (VAE) [12,13]. Subsequently, the DreamerV1-V3 [14,16,17] enhance this latent dynamics model by introducing stochastic dynamics, discrete latent representation, and symlog prediction, thus finally enabling stable application across various domains with fixed hyperparameters. Recent works [25,28,37] also incorporate the Transformer architecture into their world model, combining its robust sequence modeling and generation abilities to achieve higher performance in the Atari 100k benchmarks [8,24]. The State Space Model (SSM) is also adopted in the world model design to improve its long-term sequence modeling ability [8,24]. In addition to learning the dynamics in compact latent representation space, some works propose to design the world model as an action conditional video prediction model to generate imaginary trajectories [19, 20, 31, 32] based on advanced diffusion models [6, 29]. Despite the tremendous successes in enhancing reinforcement learning algorithms' final performance and sample efficiency, existing works on world models necessitate extensive interactions with the actual environment, limiting their efficiency.

2.2 Pre-training in World Model

To improve the learning efficiency of the world model, the pre-training and fine-tuning paradigm succeeded in computer vision and natural language processing presents a promising inspiration for this problem. To the best of our knowledge, APV [30] first innovatively studies the pre-training and fine-tuning paradigm of the world model to improve its training efficiency in downstream tasks. In the pre-training phase, APV trains an action-free latent dynamics model from action-free videos. When fine-tuning downstream tasks, another action-conditional latent dynamics model is stacked on top of the action-free latent dynamics model, and the overall model is optimized jointly to adapt to downstream tasks. The recent work, ContextWM [33], proposes an improvement of the APV framework that separately models the visual context and dynamics individually during pre-training from complex and diverse in-the-wild videos. Similar to APV, ContextWM also trains an action-free latent dynamics model in pre-training to learn state representation useful for downstream tasks but remove the interference of complex and diverse visual information irrelevant to dynamics transition. SWIM [23] pre-trains the action conditional world model in large-scale human manipulation datasets and transfers it to robotics manipulation tasks. In this study, the action labels from manipulation videos are detected explicitly using off-the-shelf visual models.

Contrary to the APV and ContextWM, which pre-trains the world model as a video predictive model without action, we pre-trains the world model with action conditional. This scheme implicitly learns the causality that links these

sequential states and keeps the same architecture between per-training and fine-tuning. As a result, it alleviates the gap between the pre-training and fine-tuning phases, enabling easy knowledge transfer to downstream tasks.

2.3 Action Representation in robotics

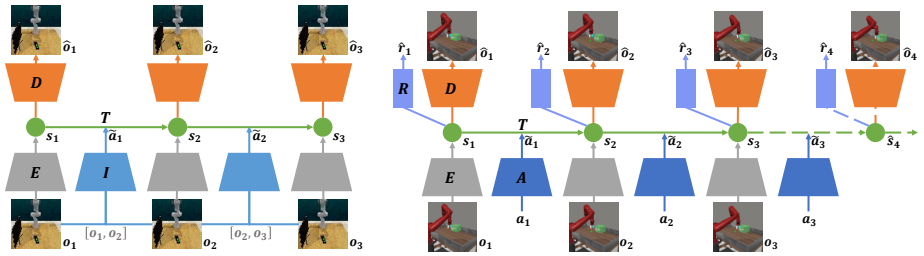
Beyond pre-training the model without considering the actions, another approach for leveraging the action-free videos is inferring the action from action-less videos and then training the world model with inferred action. In the field of robotics, VPT [4] trains an Inverse Dynamics Model (IDM) with small data labeled by humans, to create pseudo-labels for a larger dataset. Zhang et al. [36] also propose to label the video data with action predictor and use the optical flow input to facilitate the transfer of the action prediction model among datasets with different visual appearances. Ko et al. [21] estimate the action from action-less videos with the dense correspondences of optical flow and depth, using the off-the-shelf GMFlow [34] for optical flow prediction. These methods necessitate additionally labeled datasets to train the action predictor, with the accuracy of the action prediction significantly influencing the world model's performance. However, this may not fully capitalize on the potential of action-free videos. Moreover, a world model trained with pseudo-action labels is only applicable to downstream tasks with the same action space. Conversely, our method is entirely unsupervised, enabling broader usage of action-free videos without requiring labeling, thus reducing costs. Additionally, the implicit action representation in our pre-trained model broadens its application across various downstream tasks.

3 Method

3.1 Formulation

Following [30, 33], the visual-based control task in our paper is formulated as a Partially Observable Markov Decision Process (POMDP), defined as a tuple $(\mathcal{O}, \mathcal{A}, p, r, \gamma)$. $\mathcal{O}$ is the observation space, such as the high dimensional RGB image space; $\mathcal{A}$ is the action space; $p(o_t|o_{<t}, a_{<t})$ is the transition dynamics; $r(o_{\leq t}, a_{<t})$ is the reward function, and γ is the discount factor. The goal of Model-Based Reinforcement Learning (MBRL) in our formulation is to learn a policy that maximizes the expected cumulative rewards $\mathbb{E}[\sum_{t=1}^{T} \gamma^{t-1} r_t]$, where the policy is trained with the imaginary trajectories that sampled from a learned world model $(\hat{p}, \hat{r})$ approximating the environment.

In the pre-training and fine-tuning paradigm, the world model is firstly pre-trained on action-free video dataset $\mathcal{D} = \{(o_t)_{t=1}^{T}\}$ without action and reward labels, and then is fine-tuned on the trajectory data $\{o_t, a_t, r_t\}_{t=1}^{T}$ gathered from online interactions with actual environment. Leveraging the learned world model, a policy is finally developed for behavior learning of specific visual control tasks.



(a) World model pre-training **(b)** World model fine-tuning

Fig. 1: Illustration of our proposed framework. (a) The world model is pre-trained with learnable action representation from action-free videos. (b) The world model is fine-tuned for downstream tasks, e.g. manipulation. The action encoder is additionally introduced to map the real action to the implicit action representation. The uppercase letters are E: representation model, T: transition model, D: image decoder, I: inverse dynamics encoder, A: action encoder, and R: reward predictor.

3.2 World model pre-training with learnable action representation

As initiated in APV [30], three components are needed to pre-train a world model, i.e. the representation model q_ϕ that encodes observation o_t to a model state z_t, the transition model that predicts future model state $\hat{z}_t$ without access to the observation, and the image decoder that reconstructs image observations $\hat{o}_t$. The full training of the world model needs the video dataset along with action labels for each observation. However, the action-free videos used for pre-training lack action labels. Hence, in APV [30], the world model is simplified into a latent video prediction model by neglecting the action, as formulated below:

$$\begin{aligned} &\text{Representation model (APV):} && z_t \sim q_\phi(z_t|z_{t-1}, o_t), \\ &\text{Transition model (APV):} && \hat{z}_t \sim p_\phi(\hat{z}_t|z_{t-1}), \\ &\text{Image decoder (APV):} && \hat{o}_t \sim p_\phi(\hat{o}_t|z_t). \end{aligned} \tag{1}$$

As analyzed in Sec. 1, the action-free video prediction model can only capture transition dependencies between sequential states, while the final world model is action conditional which models the causality that connects the sequential states. To alleviate this discrepancy between the action-free pre-training and action-conditional fine-tuning of the world model, we instead pre-train the world model in action conditional manner. Specifically, we establish a latent dynamics model conditioned on learnable implicit action representation, as well as design an action-state consistency loss to constrain the implicit action representation closer to real action.

(a) Pre-training architecture According to our observation, humans can easily identify the actions in the videos. For example, when we see a human walking video, we can easily tell when the person in the video `'turn left'` and when they `'turn right'`. Drawing inspiration from this, we propose to learn

an implicit action $\tilde{a}_{t-1}$ from successive video frames and pre-train the world model in an action conditional manner based on this implicit action. The latent dynamics model conditioned on learnable action representation also includes three components and is formulated as follows:

$$\begin{aligned} &\text{Representation model:} && s_t \sim q_\phi(s_t|s_{t-1}, \tilde{a}_{t-1}, o_t), \\ &\text{Transition model:} && \hat{s}_t \sim p_\phi(\hat{s}_t|s_{t-1}, \tilde{a}_{t-1}), \\ &\text{Image decoder:} && \hat{o}_t \sim p_\phi(\hat{o}_t|s_t). \end{aligned} \tag{2}$$

The role of each model in Eq. (2) is the same as that in APV, but in our representation and transition model, the action is considered, i.e. in action conditional. This kind of architecture can learn the transition causality between the implicit action representation $\tilde{a}_{t-1}$ and the state $\hat{s}_t$. Moreover, the action-conditional architecture is consistent with that of the final world model. This can alleviate the discrepancy between pre-training and fine-tuning thus facilitating the fine-tuning of the world model for specific downstream tasks.

In Eq. (2), deriving a meaningful action from the action-free video is crucial for the success of pre-training. Given that sequential videos inherently contain action information, we expect to deduce this from observations at two consecutive time steps, akin to an inverse dynamics process. To achieve this, we introduce an inverse dynamics encoder designed to infer action representation from two adjacent video frames, as formulated below:

$$\tilde{a}_t \sim q_\phi(\tilde{a}_t|o_t, o_{t+1}). \tag{3}$$

(b) Pre-training loss function To optimize the model parameters in Eq. (2)-(3), two loss functions are constructed. First, a variational bound is used to maximize the data log-likelihood. Additionally, we introduce an action-state consistency loss, a novel component aiming at aligning the implicit action representation closer to the real action.

Log-likelihood maximization loss. During the pre-training phase, the model parameters ϕ in Eq. (2)-(3) are optimized by maximizing the log-likelihood of videos in the dataset $\mathcal{D}$. As shown in Eq. (4), the likelihood of the observation sequence is divided into two parts: (i) the log-likelihood of action conditional video data; and (ii) the action Kullback–Leibler (KL) loss. The latter is further decomposed according to the temporal and optimized through minimizing the evidence lower bound (ELBO). Here the decomposition is carried out based on the assumption that the current action solely depends on the observations in the current and subsequent time steps, as illustrated in Eq. (3).

$$\begin{aligned} \ln p(o_{1:T}) &\triangleq \ln \int p(o_{1:T}|\tilde{a}_{1:T}) p(\tilde{a}_{1:T}) d\tilde{a}_{1:T} \\ &\geq \mathbb{E}_{q(\tilde{a}_{1:T}|o_{1:T})}[\ln p(o_{1:T}|\tilde{a}_{1:T})] - \mathrm{KL}[q(\tilde{a}_{1:T}|o_{1:T}) \| p(\tilde{a}_{1:T})] \\ &= \mathbb{E}_{q(\tilde{a}_{1:T}|o_{1:T})}[\ln p(o_{1:T}|\tilde{a}_{1:T})] - \sum_{t=1}^{T} \mathrm{KL}[q(\tilde{a}_t|o_t, o_{t+1}) \| p(\tilde{a}_t)]. \end{aligned} \tag{4}$$

Following the derivations in [15], the Eq. (4) can be further reformulated as the following objective:

$$\mathcal{L}^{\texttt{like}}(\phi) = \mathbb{E}_{q_\phi(s_{1:T}|o_{1:T})q_\phi(\tilde{a}_{1:T}|o_{1:T})}\Big[\sum_{t=1}^{T}\Big(\underbrace{-\ln p_\phi(o_t|s_t)}_{\text{image log loss}} \tag{5}$$

$$\underbrace{+\beta \mathrm{KL}[q_\phi(s_t|s_{t-1},\tilde{a}_{t-1},o_t)\|p_\phi(\hat{s}_t|s_{t-1},\tilde{a}_{t-1})]}_{\text{dynamics KL loss}}\Big)\Big]\ \underbrace{+\beta_a \mathrm{KL}[q_\phi(\tilde{a}_t|o_t,o_{t+1})\|p(\tilde{a}_t)]}_{\text{action KL loss}},$$

where β and β_a are scale hyperparameters and T is the length of minibatch in training sequence. The first term in Eq. (5) is the same as that in the typical latent dynamics model, representing a reconstruction loss of the observation. The second term seeks to align the encoding output of the representation model with the predictions from the transition model, ensuring their closeness. The third term is to constrain actions derived from observations to obey a particular prior distribution. Here, the transition model is implemented as a recurrent state-space model (RSSM) [15] conditioned on learnable action representation to effectively capture the causality information from the action-free video.

Action-state consistency loss. The action inferred via Eq. (3) is not necessarily a good action representation due to the shortcut learning and ambiguity of implicit action space. To solve these problems, we introduce two additional constraints: a Gaussian regularization and an action-state consistency loss.

Firstly, as shown in Eq. (3), a simplistic form of shortcut learning could occur if the action representation utilizes the future observation o_{t+1} for inference, and the transition model directly employs information from $\tilde{a}_t$ (which includes the information of subsequent observation) to predict the next latent state. Such a process results in shortcut learning between action representation $\tilde{a}$ and future observation o_{t+1}, bypassing the underlying causality and leading to the collapse of the learning process within the latent dynamics model. To address the issue of shortcut learning, we constrain the distribution of the action representation to be close to the standard normal distribution by assuming $p(\tilde{a}_t)$ obey $\mathcal{N}(\mathbf{0},\mathbf{1})$ in the action KL loss in Eq. (5). This is grounded in information bottleneck theory [2], which demonstrates its effectiveness in minimizing the representation's information capacity to contain the input signals.

Secondly, to address the ambiguity of implicit action space, we introduce a self-supervised action-state consistency loss that accounts for the relationship between action and observation. For instance, if two significantly different actions are applied to the same observation, the resulting observations should also differ substantially in most cases, and vice versa. This implies that the variation in future states is positively related to the variation in actions applied to the same current state. Leveraging this insight, we formulate a specialized loss function aiming at aligning the relative differences in actions with the relative differences in their corresponding future states. This strategy is intended to bring the learnable action representation closer to the real action.

Specifically, given an initial state s_0 and two implicit action representations $\tilde{a}_1, \tilde{a}_2$, the transition model predicts two states $\hat{s}_1, \hat{s}_2$. The difference between

actions and states forms a pair:

$$< \Delta_{\tilde{a}} = \texttt{dis}(\tilde{a}_1, \tilde{a}_2), \Delta_{\hat{s}} = \texttt{dis}(\hat{s}_1, \hat{s}_2) >, \tag{6}$$

where `dis` function represents the L_2 distance when the inputs are deterministic variables and the KL divergence if the inputs are stochastic distributions.

As previously analyzed, the difference between actions (i.e.$\Delta_{\tilde{a}}$) is positively correlated to the difference in their corresponding future states (i.e. $\Delta_{\hat{s}}$). Consequently, the ordering (i.e., the result of `argsort` operation) of these differences should remain consistent within a minibatch, as illustrated below:

$$\texttt{agrsort}(\{\Delta^i_{\tilde{a}}\}^B_{i=1}) = \texttt{agrsort}(\{\Delta^i_{\hat{s}}\}^B_{i=1}), \tag{7}$$

where B is the batch size. Unfortunately, the `argsort` operator in Eq. (7) is not differentiable, making it hard to optimize. To circumvent this issue, we employ a numerical sign function as a substitute for `argsort`, which serves an equivalent purpose in our context. Consequently, we formulate the loss function as follows:

$$\mathcal{L}^r(\phi) = \frac{1}{B^2}\Big\|\texttt{sign}\Big(\Big[\|\Delta^i_{\tilde{a}} - \Delta^j_{\tilde{a}}\|_2\Big]^B_{i,j=1}\Big) - \texttt{sign}\Big(\Big[\|\Delta^i_{\hat{s}} - \Delta^j_{\hat{s}}\|_2\Big]^B_{i,j=1}\Big)\Big\|^2_2, \tag{8}$$

where the element-wise `sign` function returns 1 for input greater than 0, -1 for less than 0, and 0 for equal to 0. In our implementation, we shuffle the training data in the minibatch to construct the contrastive samples $\tilde{a}_1, \tilde{a}_2$.

Overall loss of pre-training. By summing the above objectives, the overall optimization loss in the pre-training phase is finally obtained as follows:

$$\mathcal{L}^{\texttt{pre-train}}(\phi) = \mathcal{L}^{\texttt{like}}(\phi) + \beta_r \mathcal{L}^r(\phi), \tag{9}$$

where β_r is the scale hyperparameter.

3.3 World Model Fine-tuning with Real Action

Based on the pre-trained model, we fine-tune the world model to downstream tasks using trajectory data (contains the real action) collected from the interaction with the real environment. Specifically, we introduce an action encoder mapping the real action to the implicit action representation along with a reward predictor that estimates the environmental reward. The input to the action encoder varies according to the specific visual control task at hand, allowing for diverse downstream tasks to be fine-tuned based on this pre-trained model. The overall model in fine-tuning phase is summarised in Eq. (10):

$$\begin{aligned}
&\text{Representation model:} && s_t \sim q_\phi(s_t|s_{t-1}, \tilde{a}_{t-1}, o_t),\\
&\text{Transition model:} && \hat{s}_t \sim p_\phi(\hat{s}_t|s_{t-1}, \tilde{a}_{t-1}),\\
&\text{Action encoder:} && \tilde{a}_t \sim p_\theta(\tilde{a}_t|a_t),\\
&\text{Image decoder:} && \hat{o}_t \sim p_\phi(\hat{o}_t|s_t),\\
&\text{Reward predictor:} && \hat{r}_t \sim p_\theta(\hat{r}_t|s_t).
\end{aligned} \tag{10}$$

Similar to Eq. (2), the overall model in Eq. (10) is optimized by minimizing the loss function in Eq. (11) as follows:

$$\mathcal{L}^{\texttt{fine-tune}}(\phi,\theta)=\mathbb{E}_{q_\phi(s_{1:T}|o_{1:T})p_\theta(\tilde{a}_{1:T}|a_{1:T})}\Big[\sum\nolimits_{t=1}^{T}\Big(\underbrace{-\ln p_\phi(o_t|s_t)}_{\text{image log loss}}\underbrace{-\ln p_\theta(r_t|s_t)}_{\text{reward log loss}}$$
$$\underbrace{-\ln p_\theta(r_t+r_t^{\texttt{int}}|s_t)}_{\text{auxiliary reward log loss}}\underbrace{+\beta\mathrm{KL}[q_\phi(s_t|s_{t-1},\tilde{a}_{t-1},o_t)\|p_\phi(\hat{s}_t|s_{t-1},\tilde{a}_{t-1})]}_{\text{dynamics KL loss}}\Big)\Big]. \tag{11}$$

During the optimization of the model in Eq. (10), the representation model, transition model, and image decoder are initialized with parameters from the pre-trained model. Moreover, to achieve a more effective solution, we leverage the exploration encouragement intrinsic bonus introduced in APV [30], which is formulated as follows:

$$r_t^{\texttt{int}}\doteq||\varphi(y_t)-\varphi(y_t^k)||_2, \tag{12}$$

where φ denotes the random projection [5] that maps the input to a low-dimensional representation for efficient computation of distances, and the y_t^k represents the k-nearest neighbor of y_t in a minibatch. Here we implement y_t as the observation embedding instead of the action-free model state in [30, 33] for effective computation of the intrinsic bonus.

To learn the policy to complete a specific visual control task based on the fine-tuned world model from Eq. (10), we use the actor-critic reinforcement learning algorithm for behavior learning, similar to the methodology adopted in DreamerV2 [16]. More details can be found in the supplementary materials Sec. 1.

4 Experiment and Result

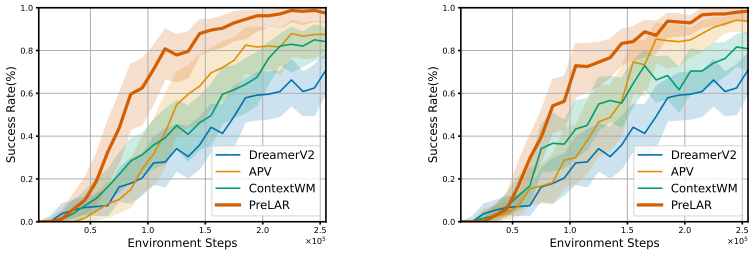
We conduct experiments in the widely used simulation environment in visual control task study to investigate the following question:

- Can the PreLAR method improve the sample efficiency of visual control tasks in robotic manipulation by pre-training from videos with learnable action representation?
- How does the PreLAR compare to the naive fine-tuning scheme?
- What is the contribution of each proposed technique in PreLAR?

Following the evaluation protocol in [1, 30], we report the interquartile mean (solid line in results figures) with bootstrap confidence interval (CI) and stratified bootstrap CI (shaded regions in results figures) for the result on individual tasks and aggregate results across 8 runs for each task.

4.1 Experiment Setup

Dataset and Simulation environment. Following the setting in APV [30], we assess the performance of PreLAR within the context of robotic manipulation tasks from the Meat-world [35] environment. For pre-training, we exploit two distinct video datasets as follows:



(a) Pre-trained on RLBench **(b)** Pre-trained on SSv2

Fig. 2: The aggregated results with pre-training on (a) RLBench and (b) SSv2. All manipulation tasks are fine-tuned and tested in the Meta-world environment.

- **RLBench** is a simulated video dataset collected from RLBench simulation environment [18] as in APV, which consists of 10 manipulation demonstrations rendered with 5 camera views in 99 tasks from RLBench, giving a total of 4950 videos.
- **Something-Something-V2 (SSv2)** is a real-world video dataset that contains 220K video clips of humans performing pre-defined, basic actions with everyday objects, such as Putting something on a surface, Moving something up, and Covering something with something [11].

For the downstream tasks, we use the Meta-world simulation environment to fine-tune the world model and test the learned manipulation behavior. In our setting, the maximum episode length is 500 steps without any action repeat, the action dimension is 4, and the reward ranges from 0-10 in all manipulation tasks.

Implementation Details. The model in the pre-training phase is trained by minimizing the objectives in Eq. (9) for 600K gradient steps. For downstream tasks, the model is fine-tuned by minimizing the objectives in Eq. (11) for 250K environment steps. We share the most hyperparameters and implementation with APV. The newly added or modified components are described below. The scale hyperparameters β, β_z, β_r in objective functions are simply set as 1.0. The 13-layer ResNets are adopted as visual encoders and decoders as in [33]. For a fair comparison, the visual encoders and decoders of APV are also replaced with 13-layer ResNets. The inverse dynamics encoder in Sec. 3.2 is also implemented as the 13-layer ResNets with visual observations of the current and subsequent frames as input. The action encoder in Sec. 3.3 is implemented as 2 layers of MLPs with 1024 hidden units and 64 output units. More details can be found in the supplementary materials Sec. 5.

4.2 Experiment Results

Results with pre-training on RLBench. Firstly, we investigate the performance of PreLAR pre-trained on the RLBench dataset and fine-tuned with ma-

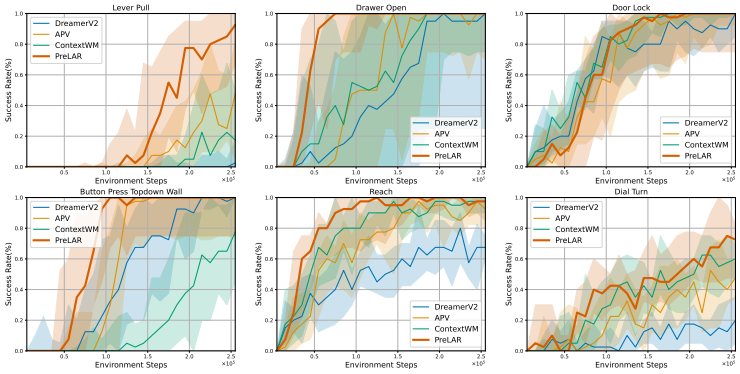


Fig. 3: The results of each manipulation task with RLBench pre-training.

nipulation tasks from the Meta-world simulated environment. The compared related works include DreamerV2, APV, and ContextWM. All pre-training methods, including APV, ContextWM, and our own, are pre-trained on the RLBench dataset and then fine-tuned on the Meta-world simulation environment. The baseline method DreamerV2 is directly trained on the online interaction trajectories with the simulation environment without pre-training.

The learning curves on six robotics manipulation tasks from the Meta-world environment and their aggregated results are shown in Fig. 3 and Fig. 2a respectively. Compared to the DreamerV2 baseline without pre-training, the pre-training approaches, APV, ContextWM, and ours PreLAR improve the final performance and sample efficiency on most tasks, indicating that per-training from action-free videos is valid for efficient world learning. Moreover, the proposed PreLAR surpasses the action-free pre-training framework APV and its improvement ContextWM. This demonstrates that the proposed scheme of learning action-conditional model during the pre-training phase alleviates the gap between pre-training and fine-tuning, thus efficiently transferring knowledge to downstream tasks. We also observe that the PreLAR outperforms other pre-training methods significantly in Drawer Open and Lever Pull tasks. We speculate that demonstrations in the training videos have a high similarity with these two tasks, hence allowing for better utilization of pre-training knowledge.

Results with pre-training on Something-Something-V2. Secondly, we further investigate the effectiveness of PreLAR when pre-trained on a dataset with a larger discrepancy, i.e. pre-training on real-world human manipulation videos (SSv2) but fine-tuning in the Meta-world simulation environment.

The learning curves on six robotics manipulation tasks from the Meta-world environment and their aggregated results are shown in Fig. 4 and Fig. 2b respectively. Similarly, we find that the pre-training approaches are generally effective in improving the final performance and sample efficiency of the world model learning, indicating the advantage of pre-training in real-world videos. Moreover,

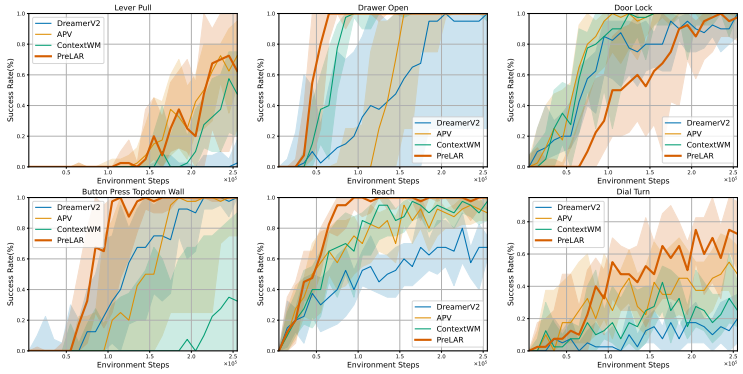
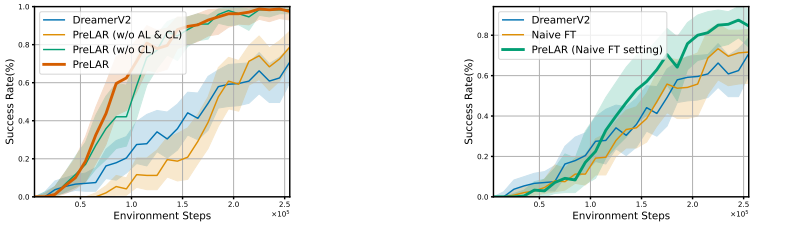


Fig. 4: The results of each manipulation task with SSv2 pre-training.



(a) Effect of proposed techniques **(b)** Comparison with naive fine-tuning

Fig. 5: The ablation results with RLBench pre-training. (a) The performance of PreLAR with removed Action KL Loss (AL) and action-state Consistency Loss (CL). (b) Comparison of our approach using an action-conditional pre-trained model against the naive fine-tuning scheme with an action-free pre-trained model.

the proposed PreLAR generally surpasses the action-free pre-training framework APV and ContextWM, demonstrating the significance of action-conditional pre-training in world model learning, even if under the challenge setting that pre-trained with real-world videos and transferred to the simulated environment. Nonetheless, the PreLAR underperforms APV on the Door Lock. For this, we guess that the large discrepancy between pre-training videos and the fine-tuning environment causes the dynamics learned in the pre-training phase to have little similarity with the Door Lock task, thus resulting in a negative transfer.

Effect of proposed techniques To evaluate the contribution of the proposed technique in PreLAR, we report the results of several variants of PreLAR by gradually removing the proposed techniques as follows:

– PreLAR (w/o CL). We set the $\beta_r = 0$ in Eq. (9) to discard the action-state Consistency Loss (CL).

- PreLAR (w/o CL & AL). We further remove the Action KL Loss (AL) in Eq. (5) and model the action representation as a deterministic variable instead of the stochastic latent variable.
- Naive Fine-Tune (FT). We do not include any of the proposed techniques and directly fine-tune the action-free pre-trained model, as in APV.

As shown in Fig. 5a, PreLAR without the action-state consistency loss (i.e. PreLAR (w/o CL)) exhibits reduced performance compared to the full PreLAR. Further, PreLAR without both the action-state consistency loss and action KL loss (i.e. PreLAR (w/o CL & AL)) exhibits a significantly reduced performance. These comparisons demonstrate the effectiveness of both losses in facilitating the learning of accurate action representations. Especially, the action KL loss appears to offer larger benefits, suggesting that the appropriate form of the implicit action representation is crucial and merits further investigation.

Furthermore, to thoroughly examine the advantages of the action-conditional pre-training strategy over the action-free pre-training strategy, we conduct a comparison between the full PreLAR and the Naive Fine-Tune scheme, within an identical architectural setup. The Naive Fine-Tune scheme fine-tunes the world model directly initialized with the action-free pre-trained model. Given that the Naive Fine-Tune scheme does not benefit from the intrinsic bonus, both the Naive Fine-Tune scheme and our PreLAR are pre-trained without the intrinsic bonus, to ensure a fair comparison.

As shown in Fig. 5b, the naive fine-tuning scheme, based on action-free pre-training, yields performance on par with the baseline DreamerV2. This outcome highlights a noticeable discrepancy between action-free pre-training and action-conditional fine-tuning. In contrast, our proposed action-conditioned pre-training scheme, PreLAR, obviously outperforms DreamerV2 and Naive FT, which clearly shows that action-conditional pre-training is advantageous for bridging the gap between pre-training and fine-tuning phases, thereby unlocking the full potential of unsupervised pre-training.

5 Conclusion and Future Work

In this work, we propose a scheme that pre-trains the world model with learnable action representation. This kind of pre-training learns implicit transition causality from action-free videos and thus facilitates the fine-tuning of the world model for specific downstream tasks, improving the efficiency of the world model and behavior learning. Our experiment results in various visual control tasks from the Meta-world environment demonstrate the effectiveness of our method in improving the sample efficiency of world model learning. However, our study acknowledges a limitation: the action representation is inferred solely from observations at two consecutive timesteps, while a more precise action representation could necessitate the consideration of a broader sequence of video frames. This also merits a direction for future research that involves exploring how to accurately learn implicit action representations from videos.

Acknowledgements

This work was partially supported by the National Science and Technology Major Project (No. 2021ZD0111901) and the Natural Science Foundation of China (Nos. U2336213 and 62122074).

References

1. Agarwal, R., Schwarzer, M., Castro, P.S., Courville, A.C., Bellemare, M.: Deep reinforcement learning at the edge of the statistical precipice. In: Advances in Neural Information Processing Systems (NeurIPS). vol. 34, pp. 29304–29320 (2021)
2. Alemi, A.A., Fischer, I., Dillon, J.V., Murphy, K.: Deep variational information bottleneck. In: International Conference on Learning Representations (ICLR) (2017)
3. Allen, K.R., Smith, K.A., Tenenbaum, J.B.: Rapid trial-and-error learning with simulation supports flexible tool use and physical reasoning. Proceedings of the National Academy of Sciences (PNAS) **117**(47), 29302–29310 (2020)
4. Baker, B., Akkaya, I., Zhokov, P., Huizinga, J., Tang, J., Ecoffet, A., Houghton, B., Sampedro, R., Clune, J.: Video pretraining (vpt): Learning to act by watching unlabeled online videos. In: Advances in Neural Information Processing Systems (NeurIPS) (2022)
5. Bingham, E., Mannila, H.: Random projection in dimensionality reduction: applications to image and text data. In: ACM International Conference on Knowledge Discovery and Data Mining (KDD). pp. 245–250 (2001)
6. Blattmann, A., Rombach, R., Ling, H., Dockhorn, T., Kim, S.W., Fidler, S., Kreis, K.: Align your latents: High-resolution video synthesis with latent diffusion models. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 22563–22575 (2023)
7. Daniel, K.: Thinking, fast and slow. Macmillan (2011)
8. Deng, F., Park, J., Ahn, S.: Facing off world model backbones: Rnns, transformers, and s4. In: Oh, A., Naumann, T., Globerson, A., Saenko, K., Hardt, M., Levine, S. (eds.) Advances in Neural Information Processing Systems (NeurIPS). vol. 36, pp. 72904–72930 (2023)
9. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An image is worth 16x16 words: Transformers for image recognition at scale. In: International Conference on Learning Representations (ICLR) (2021)
10. Du, Y., Yang, S., Florence, P., Xia, F., Wahid, A., brian ichter, Sermanet, P., Yu, T., Abbeel, P., Tenenbaum, J.B., Kaelbling, L.P., Zeng, A., Tompson, J.: Video language planning. In: International Conference on Learning Representations (ICLR) (2024)
11. Goyal, R., Kahou, S.E., Michalski, V., Materzynska, J., Westphal, S., Kim, H., Haenel, V., Fruend, I., Yianilos, P., Mueller-Freitag, M., Hoppe, F., Thurau, C., Bax, I., Memisevic, R.: The "something something" video database for learning and evaluating visual common sense. In: IEEE International Conference on Computer Vision (ICCV). pp. 5843–5851 (2017)
12. Ha, D., Schmidhuber, J.: Recurrent world models facilitate policy evolution. In: Advances in Neural Information Processing Systems (NeurIPS). vol. 31 (2018)
13. Ha, D., Schmidhuber, J.: World models. arXiv preprint arXiv:1803.10122 (2018)

14. Hafner, D., Lillicrap, T., Ba, J., Norouzi, M.: Dream to control: Learning behaviors by latent imagination. In: International Conference on Learning Representations (ICLR) (2020)
15. Hafner, D., Lillicrap, T., Fischer, I., Villegas, R., Ha, D., Lee, H., Davidson, J.: Learning latent dynamics for planning from pixels. In: International Conference on Machine Learning (ICML). pp. 2555–2565 (2019)
16. Hafner, D., Lillicrap, T.P., Norouzi, M., Ba, J.: Mastering atari with discrete world models. In: International Conference on Learning Representations (ICLR) (2021)
17. Hafner, D., Pasukonis, J., Ba, J., Lillicrap, T.: Mastering diverse domains through world models. arXiv preprint arXiv:2301.04104 (2023)
18. James, S., Ma, Z., Arrojo, D.R., Davison, A.J.: Rlbench: The robot learning benchmark & learning environment. IEEE Robotics and Automation Letters (RAL) **5**(2), 3019–3026 (2020)
19. Jia, F., Mao, W., Liu, Y., Zhao, Y., Wen, Y., Zhang, C., Zhang, X., Wang, T.: Adriver-i: A general world model for autonomous driving. arXiv preprint arXiv:2311.13549 (2023)
20. Kaiser, Ł., Babaeizadeh, M., Miłos, P., Osiński, B., Campbell, R.H., Czechowski, K., Erhan, D., Finn, C., Kozakowski, P., Levine, S., Mohiuddin, A., Sepassi, R., Tucker, G., Michalewski, H.: Model based reinforcement learning for atari. In: International Conference on Learning Representations (ICLR) (2020)
21. Ko, P.C., Mao, J., Du, Y., Sun, S.H., Tenenbaum, J.B.: Learning to act from actionless videos through dense correspondences. In: International Conference on Learning Representations (ICLR) (2024)
22. LeCun, Y.: A path towards autonomous machine intelligence. Open Review **62** (2022)
23. Lu, C., Schroecker, Y., Gu, A., Parisotto, E., Foerster, J.N., Singh, S., Behbahani, F.: Structured state space models for in-context reinforcement learning. In: ICML workshop on new frontiers in learning, control, and dynamical systems (2023)
24. Mattes, P., Schlosser, R., Herbrich, R.: Hieros: Hierarchical imagination on structured state space sequence world models. arXiv preprint arXiv:2310.05167 (2023)
25. Micheli, V., Alonso, E., Fleuret, F.: Transformers are sample-efficient world models. In: International Conference on Learning Representations (ICLR) (2023)
26. Moerland, T.M., Broekens, J., Plaat, A., Jonker, C.M., et al.: Model-based reinforcement learning: A survey. Foundations and Trends® in Machine Learning **16**(1), 1–118 (2023)
27. Radford, A., Narasimhan, K., Salimans, T., Sutskever, I., et al.: Improving language understanding by generative pre-training (2018), `https://www.mikecaptain.com/resources/pdf/GPT-1.pdf`
28. Robine, J., Höftmann, M., Uelwer, T., Harmeling, S.: Transformer-based world models are happy with 100k interactions. In: International Conference on Learning Representations (ICLR) (2023)
29. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 10684–10695 (2022)
30. Seo, Y., Lee, K., James, S.L., Abbeel, P.: Reinforcement learning with action-free pre-training from videos. In: International Conference on Machine Learning (ICML). pp. 19561–19579 (2022)
31. Wang, X., Zhu, Z., Huang, G., Chen, X., Lu, J.: Drivedreamer: Towards real-world-driven world models for autonomous driving. arXiv preprint arXiv:2309.09777 (2023)

32. Wang, Y., He, J., Fan, L., Li, H., Chen, Y., Zhang, Z.: Driving into the future: Multiview visual forecasting and planning with world model for autonomous driving. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 14749–14759 (2024)
33. Wu, J., Ma, H., Deng, C., Long, M.: Pre-training contextualized world models with in-the-wild videos for reinforcement learning. In: Advances in Neural Information Processing Systems (NeurIPS). vol. 36, pp. 39719–39743 (2023)
34. Xu, H., Zhang, J., Cai, J., Rezatofighi, H., Tao, D.: Gmflow: Learning optical flow via global matching. In: IEEE/CVF conference on Computer Vision and Pattern Recognition (CVPR). pp. 8121–8130 (2022)
35. Yu, T., Quillen, D., He, Z., Julian, R., Hausman, K., Finn, C., Levine, S.: Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning. In: Conference on Robot Learning (CoRL) (2019)
36. Zhang, Q., Peng, Z., Zhou, B.: Learning to drive by watching youtube videos: Action-conditioned contrastive policy pretraining. In: European Conference on Computer Vision (ECCV). pp. 111–128 (2022)
37. Zhang, W., Wang, G., Sun, J., Yuan, Y., Huang, G.: Storm: Efficient stochastic transformer based world models for reinforcement learning. In: Advances in Neural Information Processing Systems (NeurIPS) (2023)