frontiers | Frontiers in Neurorobotics

TYPE Original Research
PUBLISHED 30 April 2025
DOI 10.3389/fnbot.2025.1585386

OPEN ACCESS

EDITED BY
Xianmin Wang,
Guangzhou University, China

REVIEWED BY
Shuai Dong,
University of Electronic Science and Technology of China Zhongshan Institute, China
Siyu Dai,
Amazon, United States
Xin Li,
Tongji University, China

*CORRESPONDENCE
Stefano Ferraro
✉ stefano.ferraro@ugent.be

†These authors have contributed equally to this work

RECEIVED 28 February 2025
ACCEPTED 01 April 2025
PUBLISHED 30 April 2025

CITATION
Ferraro S, Mazzaglia P, Verbelen T and Dhoedt B (2025) FOCUS: object-centric world models for robotic manipulation. *Front. Neurorobot.* 19:1585386. doi: 10.3389/fnbot.2025.1585386

FOCUS: object-centric world models for robotic manipulation

Stefano Ferraro[1*†], Pietro Mazzaglia[1†], Tim Verbelen[2] and Bart Dhoedt[1]

[1]IDLab, Department of Information Technology, Ghent University–imec, Ghent, Belgium, [2]VERSES Research Lab, VERSES, Los Angeles, CA, United States

Understanding the world in terms of objects and the possible interactions with them is an important cognitive ability. However, current world models adopted in reinforcement learning typically lack this structure and represent the world state in a global latent vector. To address this, we propose FOCUS, a model-based agent that learns an object-centric world model. This novel representation also enables the design of an object-centric exploration mechanism, which encourages the agent to interact with objects and discover useful interactions. We benchmark FOCUS in several robotic manipulation settings, where we found that our method can be used to improve manipulation skills. The object-centric world model leads to more accurate predictions of the objects in the scene and it enables more efficient learning. The object-centric exploration strategy fosters interactions with the objects in the environment, such as reaching, moving, and rotating them, and it allows fast adaptation of the agent to sparse reward reinforcement learning tasks. Using a Franka Emika robot arm, we also showcase how FOCUS proves useful in real-world applications. Website: `focus-manipulation.github.io`.

KEYWORDS

world models, object-centric representation, neuro robotics, object-centric exploration, embodied-AI

1 Introduction

In our daily lives, we effortlessly interact with objects to accomplish a wide range of tasks. Through these interactions, we instinctively infer an object's identity, spatial position, 3D structure, appearance, and texture, effectively building a generative model of how objects are formed (Parr et al., 2021). For robot manipulators, replicating these tasks presents a significant challenge due to the intricate and dynamic nature of interactions between the agent and its environment.

In recent years, deep reinforcement learning (RL) has shown to be a promising approach for dealing with complex manipulation scenarios (Levine et al., 2016; OpenAI et al., 2019; Kalashnikov et al., 2018; Lu et al., 2021; Lee et al., 2021; Ferraro et al., 2022a). Among RL algorithms, model-based approaches aspire to provide greater data efficiency, compared to the model-free counterparts (Fujimoto et al., 2018; Haarnoja et al., 2018). Adopting world models (Ha and Schmidhuber, 2018; Hafner et al., 2021), i.e. generative models that learn the environment's dynamics by reconstructing sensory observations, model-based agents have shown impressive performance across several domains (Hafner et al., 2021; Rajeswar et al., 2023; Hafner et al., 2023), including real-world applications, such as robotic manipulation and locomotion (Wu et al., 2022). However, world models that indistinctly reconstruct all information in the environment can suffer from several failure modes. For instance, in visual tasks, they can ignore small, but important features

for predicting the future, such as small objects (Seo et al., 2022). They also tend to waste the model capacity on visually rich, but irrelevant features, such as static backgrounds (Deng et al., 2022). In the case of robotic manipulation, this is problematic because the agent strongly needs to acquire information about the objects to manipulate to solve a given task.

Another challenge in RL for manipulation is engineering reward functions that drive the agent's learning toward task completion. Attempting to design dense reward functions easily leads to faulty reward designs (Amodei et al., 2016; Clark and Amodei, 2016; Krakovna et al., 2020; Popov et al., 2017). One solution is to adopt sparse reward feedback, providing a positive reward only for successful task completion. However, these functions are challenging to optimize with RL, due to the difficulty of finding such rewards in the environment. Thus, they require appropriate exploration strategies, for which previous work has resorted to artificial curiosity mechanisms (Oudeyer et al., 2007; Schmidhuber, 1991) or entropy maximization strategies (Mutti et al., 2021; Liu and Abbeel, 2021). In Liu and Abbeel (2021), exploration emerges by maximizing the entropy over the full latent representation, resulting in the agent potentially focusing on exploring irrelevant aspects of the scene (Burda et al., 2018b).

Humans, on the other hand, tend to develop a structured mental model of the world by interacting with objects registering specific features associated with objects, such as shape, color, etc. (Hawkins et al., 2017; Ferraro et al., 2023). Since infancy, toddlers learn this by actively engaging with objects and manipulating them with their hands, discovering object-centric views that allow them to build an accurate mental model (Smith et al., 2018; Slone et al., 2019; Ferraro et al., 2022b).

In this work, we present an approach inspired by the principle that objects should be of primary importance in an agent's world model, and motivated by the above issues regarding: i) the complexity of modeling object entities in the environment, and ii) the necessity of autonomously discovering interactions with such objects. We introduce **FOCUS**, a model-based RL agent that learns an object-centric representation of the world. Unlike holistic scene representations, an object's latent vector allows the agent to prioritize information about objects. Leveraging the object-centric representation, it's possible to design an exploration strategy that focuses on the interactions where objects are involved. Crucially, the proposed focused exploration strategy allows for improved performance on sparsely rewarded tasks when compared to the state-of-the-art.

Our contributions in this work can be summarized as:

- an object-centric world model, which learns the latent dynamics of the environment where the information about objects is discriminated into distinct latent vectors;
- an object-centric exploration strategy, which encourages interactions with the objects, by maximizing the entropy of the latent object's representation;
- empirical evaluation of the approach, showing how object-centric models improve the agent's understanding of the objects in the scene and how the object-centric exploration strategy fosters interaction with the objects. This leads the agent to more efficiently solve robotic manipulation tasks in several settings and tasks, including ManiSkill2 (Gu et al., 2023), robosuite (Zhu et al., 2020) and Metaworld (Yu et al., 2019) environments.
- a deployment on a real robotic platform, showcasing the possibility of successfully applying our approach to a hardware-based setup.

2 Background

2.1 Reinforcement learning and world models

In RL, the agent receives inputs x from the environment and can interact through actions a. The objective of the agent is to maximize the discounted sum of rewards $\sum_t \gamma^t r_t$, where t indicates discrete timesteps. In order to do so, RL agents learn an optimal policy $\pi(a|x)$ outputting actions that maximize the expected cumulative discounted reward over time, generally estimated using a critic function, which can be either a state-value function or an action-value function (Haarnoja et al., 2018; Fujimoto et al., 2018). In addition, model-based RL methods learn a model of the transition dynamics of the environment and use it to select actions (Hansen et al., 2022) or to optimize the actor-critic networks (Janner et al., 2021). Recently, world models (Ha and Schmidhuber, 2018) have adopted deep generative models (Goodfellow et al., 2016) to learn the dynamics of the environment, capturing the environment dynamics into a latent space, which can be used to learn the actor and critic functions using imaginary rollouts (Hafner et al., 2021, 2023) or to actively plan at each action (Schrittwieser et al., 2020; Rajeswar et al., 2023; Song et al., 2024). Given that world model-based RL has been shown to be more efficient than model-free RL (Hafner et al., 2023) and the importance of sample-efficiency in robotic manipulation, we base our work on world models and RL for learning behaviors from interactions.

2.2 Exploration

Solving sparse-reward tasks is a hard problem in RL because of the difficulty of exploring the environment and identifying rewarding states. Inspired by artificial curiosity theories (Schmidhuber, 1991; Oudeyer et al., 2007), several works have designed exploration strategies for RL (Pathak et al., 2017; Mazzaglia et al., 2022; Rajeswar et al., 2021). Other exploration strategies that have shown great success are based upon the ideas of maximizing uncertainty (Pathak et al., 2019; Sekar et al., 2020), or the entropy of the agent's state representation (Liu and Abbeel, 2021; Seo et al., 2021; Mutti et al., 2021). One issue with exploration in visual environments is that these approaches can be particularly attracted by easy-to-reach states that strongly change the visual appearance of the environment (Burda et al., 2018a). In robotic manipulation, this can cause undesirable behaviors, e.g., a robot arm exploring different poses in the proximity of the camera but ignoring interactions

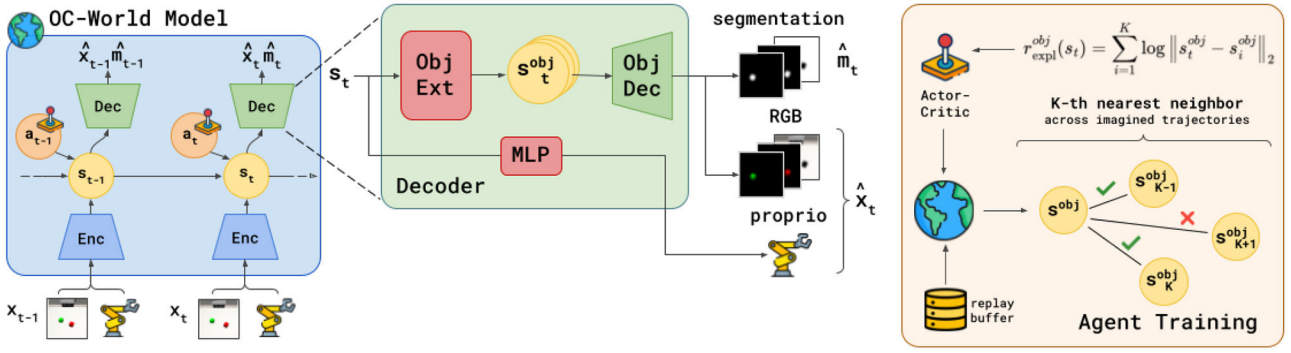


FIGURE 1
Overview of FOCUS. The agent learns a structured world model **(left)** that disentangles information in the environment by learning to reconstruct masked information about each observed object, thanks to an object-centric decoder **(center)**. The learned object-centric state representation is used to train a policy that incentivizes object-centric exploration **(right)**, maximizing the entropy of the object representation as a form of intrinsic reward.

with the objects in the workspace (Rajeswar et al., 2023). Our method, instead, leverages object-centric representations to encourage agents to interact with the objects present in the scene. By designing an object-centric exploration strategy, we provide a better alternative to curiosity mechanisms for robotic manipulation, which have no specific targets for exploration in the environment.

2.3 Object-centric representations

Decomposing scenes into objects can enable efficient reasoning over high-level building blocks and ensure the agent focuses on the most relevant concepts (Dittadi et al., 2021). Several 2D object-centric representations, based on the principle of representing objects as separate entities within the model, have recently emerged (Locatello et al., 2020; Greff et al., 2020; Burgess et al., 2019; Nakano et al., 2023). Due to computational and quality constraints, these object-centric representations have not been extended to more complex scenarios, where the interaction with an agent is also to be modeled. Related work investigated the usefulness of object-centric representations for control, using model-free RL (Diuk et al., 2008; Janner et al., 2019; Kipf et al., 2020; Yoon et al., 2023). Inspired by these approaches, we propose an object-centric world model which allows us to learn behaviors efficiently by leveraging model-based RL. The object-centric representation improves the agent's predictions about objects and can be used both to enable more accurate control, e.g. to solve dense-rewards RL tasks, and to foster interactions with objects using a new object-centric exploration strategy, e.g. in sparse-rewards RL tasks. The closest work in literature to our approach (Sancaktar et al., 2022) is an object-centric exploration strategy based on graph-structured models for control. However, this approach requires already precise information about objects, e.g. the position, which is generally available only in simulation. Instead, our approach is designed to work well for common visual manipulation settings, where information about the scene is provided to the agent only through camera images.

3 Object-centric world model

The agent observes the environment through the inputs $x_t = \{o_t, q_t\}$ it receives at each interaction, where we can distinguish the (visual) observations o_t, e.g. camera RGB, from the proprioceptive information q_t, e.g. the robot joint states and velocities. This information is processed by the agent through an encoder model $e_t = f(x_t)$, which can be instantiated as the concatenation of the outputs of a CNN for high-dimensional observations and an MLP for low-dimensional proprioception.

The world model aims to capture the dynamics of the inputs into a latent state s_t. In previous work (Hafner et al., 2021), this is achieved by reconstructing the inputs using an observation decoder. With FOCUS, we are interested in separating object-specific information into separate latent representations s_t^{obj}. For this reason, we instantiate two object-conditioned components: an *object latent extractor* and an *object decoder*. We first describe the structure and loss of the world model (in Figure 1, **left**) before delving into more details about the novel object-centric components of FOCUS (in Figure 1, **center**).

3.1 World model

Overall, the learned world model is composed of the following components:

$$\begin{aligned}
\text{Encoder:} \quad & e_t = f(x_t), \\
\text{Posterior:} \quad & p_\phi(s_{t+1} | s_t, a_t, e_{t+1}), \\
\text{Prior:} \quad & p_\phi(s_{t+1} | s_t, a_t), \\
\text{Proprio decoder:} \quad & p_\theta(\hat{q}_t | s_t), \\
\text{Object latent extractor:} \quad & p_\theta(s_t^{obj} | s_t, c^{obj}), \\
\text{Object decoder:} \quad & p_\theta(\hat{o}_t^{obj}, l_t^{obj} | s_t^{obj}).
\end{aligned}$$

which are trained end-to-end by minimizing the following loss:

$$\mathcal{L}_{\text{wm}} = \mathcal{L}_{\text{dyn}} + \mathcal{L}_{\text{proprio}} + \mathcal{L}_{\text{obj}}. \quad (1)$$

We explain each component in details in the following paragraphs.

For the dynamics component, i.e., prior and posterior, we adopt a recurrent state-space model (RSSM) (Hafner et al., 2019), which extracts a latent state s_t made of a deterministic and a stochastic component. The parameters of the RSSM modules are collectively denoted as ϕ. The dynamics minimize the Kullback-Leibler (KL) divergence between posterior and prior:

$$\mathcal{L}_{\text{dyn}} = D_{\text{KL}}[p_\phi(s_{t+1}|s_t, a_t, e_{t+1})||p_\phi(s_{t+1}|s_t, a_t)]. \quad (2)$$

All parameters of the decoding units of the network are represented by θ. Proprioceptive information $\hat{q}_t$ is decoded out of the latent state s_t, using an MLP. The proprioceptive decoder learns to reconstruct proprio states, by minimizing a negative log-likelihood (NLL) loss:

$$\mathcal{L}_{\text{proprio}} = -\log p_\theta(\hat{q}_t|s_t) \quad (3)$$

3.2 Object-centric modules

The latent state of the world model tends to compress all the information from the environment in a unique latent structure. Our intention in FOCUS is to disentangle such information into separate latent structures, learning an object-centric world model.

For each object in the scene, the *object latent extractor* receives the model latent state s_t and a (one-hot) vector identifying the object c^{obj}, and extracts an object-centric latent s_t^{obj}. Given such an object latent, the *object decoder* reconstructs object-related observation information by outputting two kinds of information: one-dimensional "object logits" l_t^{obj}, which are used to build a segmentation mask of the scene, and object-specific observation $\hat{o}_t^{obj}$, where the information that is irrelevant to the object is masked out through the segmentation. *How is the segmentation mask learned?* The object decoder outputs one-dimensional "object logits" l_t^{obj}, which represent object-specific per-pixel logits. These logits are aggregated in a scene by applying a softmax among all object weights. The overall segmentation mask is obtained as:

$$\hat{m}_t = \text{softmax}(l_t^1, ..., l_t^N) \quad (4)$$

with N being the object instances. Object-specific masks can be obtained by taking the corresponding object's channel mask in the segmentation. Defining object-specific masks as m_t^{obj}, we can multiply the observation by these masks, to obtain object-specific observations $\hat{o}_t^{obj}$ that focus only on the *obj*-th object information. [1]

The object decoder loss is defined as follows:

$$\mathcal{L}_{\text{obj}} = -\log \underbrace{p(\hat{m}_t)}_{\text{mask}} - \log \sum_{\text{obj}=0}^{N} \underbrace{m_t^{\text{obj}} p_\theta(\hat{x}_t^{\text{obj}}|s_t^{\text{obj}})}_{\text{masked reconstruction}} \quad (5)$$

By minimizing the NLL of the masked reconstruction term, the object-decoder ensures that each object latent s^i focuses on capturing only its relevant information, as the reconstructions obtained from the latent are masked per object. Furthermore, objects compete to occupy their correct space in the scene (in pixel space), through the *mask* loss.

How are the segmentation mask targets for the mask loss obtained? In order to discriminate object information into different latent vectors, the object-centric components leverage an object discrimination process that entails learning to segment the scene observations. Some simulated robotic environments make this information available, however, the same process is non-trivial in real-world settings.

The increasing availability of large pre-trained models for segmentation offers an opportunity to avoid the problem. Thus, in our experiments, we adopt an efficient implementation of the Segment Anything Model (fastSAM; Kirillov et al., 2023; Zhao et al., 2023). At the beginning of each episode, per object segmentation instances are generated with fastSAM, using box or text prompts. For subsequent frames, segmentation maps are produced by a tracking model, for which we ground on the XMem model (Yang et al., 2023). This strongly simplifies the process of obtaining segmentation masks in robotic workspaces.

4 Object-centric exploration

State maximum entropy approaches for RL (Mutti et al., 2021; Seo et al., 2021; Liu and Abbeel, 2021) learn an environment representation, on top of which they compute an entropy estimate that is maximized by the agent's actor to foster exploration. Given our object-centric representation, we can incentivize well-directed exploration toward object interactions and the discovery of novel object views, by having the agent maximize the entropy over the object latent state representation.

In order to estimate the entropy value over batches, we apply a K-NN particle-based estimator (Singh et al., 2003) on top of the object latent representation. By maximizing the overall entropy, with respect to all objects in the scene, we derive the following reward for object-centric exploration:

$$r_{\text{expl}} = \sum_{obj=0}^{N} r_{\text{expl}}^{obj}$$
$$\text{where} \quad r_{\text{expl}}^{obj}(s) \propto \sum_{i=1}^{K} \log \left\| s^{obj} - s_i^{obj} \right\|_2 \quad (6)$$

where s^{obj} is extracted from s using the object latent extractor, s_i^{obj} is the i-th nearest neighbor to s^{obj}.

Crucially, as we learn an (object-centric) world model we can use it to optimize actions by learning actor and critic in imagination (Hafner et al., 2021), so that the latent states in Equation 6 are states of imaginary trajectories, generated by the world model by following the actor's predicted actions.

Learning actor-critic in imagination allows one to efficiently learn actions by generating hypothetical trajectories in the agent's latent state space. This can be done by applying RL for learning an actor policy $\pi(a_t|s_t)$ that outputs actions that maximize the following bootstrapped λ-returns (Hafner et al., 2023):

$$R_t^\lambda = r_t + \gamma((1-\lambda)v(s_{t+1}) + \lambda R_{t+1}^\lambda) \quad (7)$$

1 The scene, with objects masked out, is also considered a "special object". N is a chosen parameter, related to the objects of interest in the scene.

with the value function $v(s_t)$ learning to approximate R_t^λ.

Given the above, we can learn an exploration actor-critic:

$$\text{Exploration actor:} \quad \pi_{\text{expl}}(a_t|s_t), \qquad \text{Exploration critic:} \quad v_{\text{expl}}(s_t), \tag{8}$$

that learns to maximize the exploration reward in Equation 6.

FOCUS acts at two levels: it explores to find useful interactions, and consequently it learns to perform downstream tasks using the sparse rewards found in the environment.

Indeed, as the agent explores the environment, it may encounter important information that may be a source of (sparse) reward, (e.g. opening a drawer). To exploit such information, while we keep exploring, we concurrently train a *task* reward predictor $r_{\text{task}}(s_t)$ and actor-critic, which can be used for solving the pre-defined task after exploring the environment, in a zero-shot or few-shot fashion.

The task actor-critic is defined as follows:

$$\text{Task actor:} \quad \pi_{\text{task}}(a_t|s_t), \qquad \text{Task critic:} \quad v_{\text{task}}(s_t). \tag{9}$$

and it is trained by maximization of the expected reward predicted. Thanks to the world model, the reward is inferred in imagination, so the learning of the task actor-critic can happen fully in imagination, while the agent keeps exploring the environment (Sekar et al., 2020).

5 Experiments

We argue that the FOCUS object-centric world model and exploration strategy can be used to improve control in robotic manipulation, especially in sparse-reward settings. The experiments aim to empirically validate our argument by evaluating (i) the exploration performance of FOCUS compared to the state-of-the-art in world models and exploration, (ii) performance on sparse reward manipulation tasks, after an exploration stage. (iii) We validate the performance of the object-centric world model on dense reward tasks and present an additional analysis of the model, e.g. visualizing the reconstructions of the world model. Finally, we deploy FOCUS to a real-world setup.

5.1 Exploration-adaptation in sparse-reward tasks

We adopt 10 tasks from three robotic manipulation benchmarks (shown in Figure 2): ManiSkill2 (Gu et al., 2023), robosuite (Zhu et al., 2020) and Metaworld (Yu et al., 2019). Both ManiSkill and robosuite provide segmentation masks as an (optional) input for the agent, while Metaworld does not. Thus, we adopted fastSAM (Zhao et al., 2023) to extract segmentation masks in those tasks, an evaluation setting that serves us the purpose of a test field for real-world experiments. The object of interest is prompted using text (Cheng et al., 2023), providing the name of the object in the scene. The masking produced by the SAM model is treated as the object masking, while the negative of it as background masking.

We compare FOCUS against three exploration strategies: Plan2Explore (P2E) (Sekar et al., 2020), Active Pre-training (APT) (Liu and Abbeel, 2021) and Random actions. For fairness with P2E and FOCUS, both APT and Random are implemented on top of a DreamerV2 world-model-based agent, following (Rajeswar et al., 2023) and using their open-source code. The hyperparameters are the same used for DreamerV2 (Hafner et al., 2021), with the exception of the batch size and sequence length, both equal to 32.

For the implementation of FOCUS, we introduced an object latent extractor unit consisting of a 3-layer MLP with a dimensionality of 512. The object-decoder network resembles the structure of the Dreamer's decoder, the depth factor for the CNN is set to 72. The K-NN filter adopted for the entropy approximation uses a K-nearest neighbors factor of K = 30.

5.1.1 Exploration

To compare the performance of different exploration strategies for manipulation, we chose a set of metrics that are related to interactions with objects:

- *Contact (%):* average percentage of contact interactions between the gripper and the objects over an episode.
- *Positional displacement (m):* cumulative position displacement of all the objects over an entire episode.
- *Angular displacement (rad):* cumulative angular displacement of all the objects over an entire episode.

In Figure 3, we observe that FOCUS interacts with objects much more assiduously than the other approaches, with the exploration performance consistently increasing over time. APT and P2E perform similarly and they only slightly perform better than Random, showing the importance of focussing on objects when exploring a robotic manipulation environment.

5.1.2 Sparse reward tasks fine-tuning

During the exploration stage, all agents explore different actions in the environment, discovering the dynamics and reward function for a given task. However, the agents make no use of the task rewards during the exploration stage. After exploring the environment for 2M environment steps, we adapt the task actor-critic, using the rewards found during the exploration stage and allowing an additional (smaller) number of environment interactions for fine-tuning the agent and perfecting the task. The adaptation curves for six tasks, showing episode rewards over time, are presented in Figure 4.

The results show that FOCUS is the method that makes the most significant progress across all tasks. This proves that the agent consistently found sparse rewards in the environment, making adaptation to a given task easier. In support of this hypothesis, the fine-tuning performance starts increasing almost immediately in all tasks despite the sparse nature of the rewards. As for the other methods, we observe that Plan2Explore and APT were able to consistently find rewarding interactions only in a few tasks (Drawer Open, Door Close), where they perform well and similarly to FOCUS. Given a sparse reward signal, and not a dense one, it makes it hard for the methods with a limited exploration strategy to

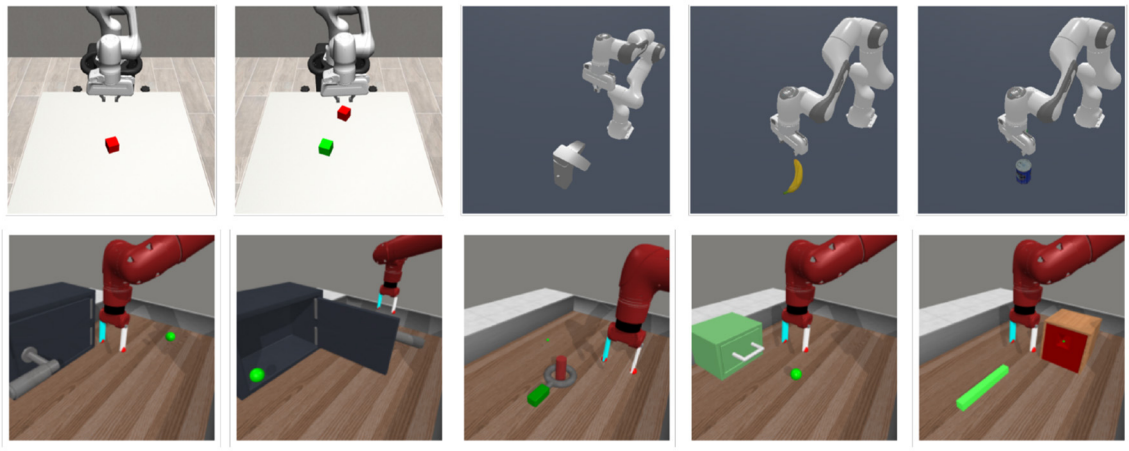


FIGURE 2
Manipulation environments. The 10 tasks we adopted are part of ManiSkill2 (MS), robosuite (RS) and Metaworld (MW). From **left–right**: Red cube (RS), RG cubes (RS), Faucet (MS), Banana (MS), Master Chef Can (MS), Door Open (MW), Door Close (MW), Disassemble (MW), Drawer Open (MW), Peg Insert (MW).

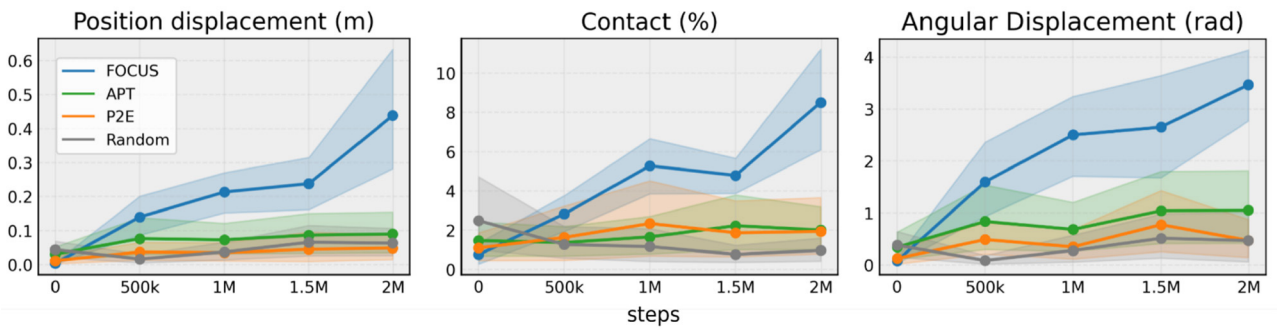


FIGURE 3
Exploration performance. Comparing exploration metrics across 10 tasks from ManiSkill2, robosuite and Metaworld. Experiments are run with three seeds per task and aggregated in a statistically sound way using RLiable (Agarwal et al., 2021).

achieve good performance when deployed for fine-tuning. Instead, Random, being the most naive exploration strategy, barely found any rewards, making fine-tuning in sparse rewards settings difficult.

5.2 Additional analysis

We have developed object-centric world models to improve the way objects' information is represented in the world model, by leveraging a structured latent representation. We perform an additional analysis, to show that objects' prediction improves when employing object-centric structured world models, compared to using a "flat" latent structure, and to validate the information contained in the latent object states.

5.2.1 Comparison to "flat" world models

Objects' size in the workspace is generally smaller than other elements, e.g. the robot, the table, and the background. When using a "flat" representation of the world, as Dreamer (Hafner et al., 2021) does, visual information about objects might be lost in the compression due to the encoding-decoding process of the world model. Qualitative reconstructions from the decoder of FOCUS are compared to reconstructions of Dreamer in Figure 5. Thanks to the explicit object's modeling FOCUS is able to reconstruct accurately any objects in the scene. Dreamer fails in many of these scenarios, especially in case of small objects, with poor visual contrast with respect to the background. In both Master Chef Can and Banana environments, Dreamer approximates each object in the scene with a cloudy presence, reflecting the lack of significant error signal to achieve a detailed reconstruction. To quantify the different performances in objects' predictions, we show the prediction error in the image area surrounding objects, in Figure 6. FOCUS is consistent in delivering more accurate object predictions.

Do better object predictions yield better manipulation performance? In order to isolate the problem of learning manipulation tasks from exploration, we compare FOCUS and Dreamer performance on a set of six dense-reward tasks: Drawer Open, Door Open, Door Close, Lift Cube, Stack Cube, and Turn Faucet. This comparison allows us to determine

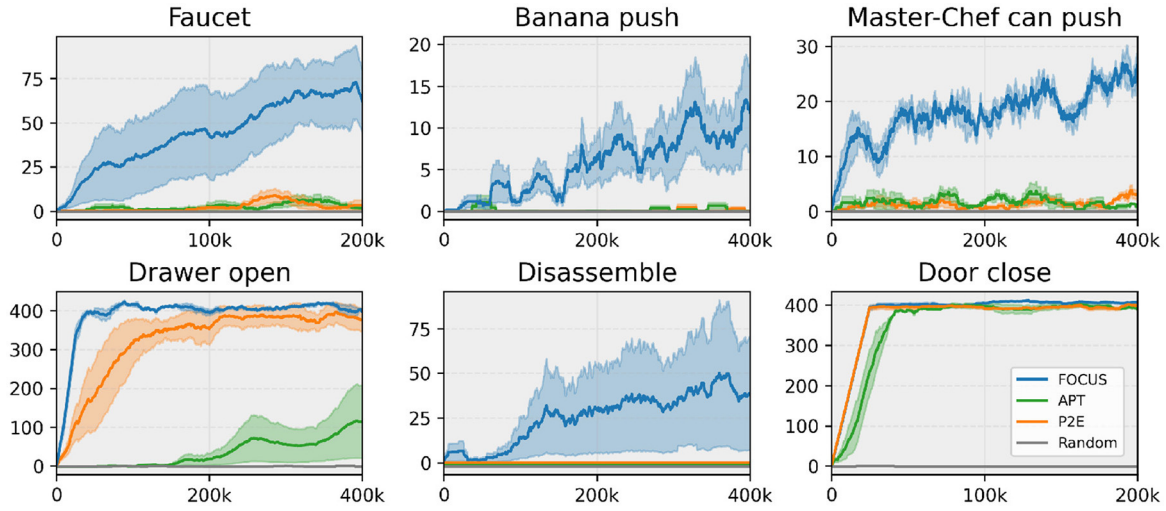


FIGURE 4
Sparse task fine-tuning performance. Comparing fine-tuning performance across tasks from ManiSkill2 (Faucet, Banana, Master Chef Can) and Metaworld (Drawer, Disassemble, Door). Experiments are run with three seeds.

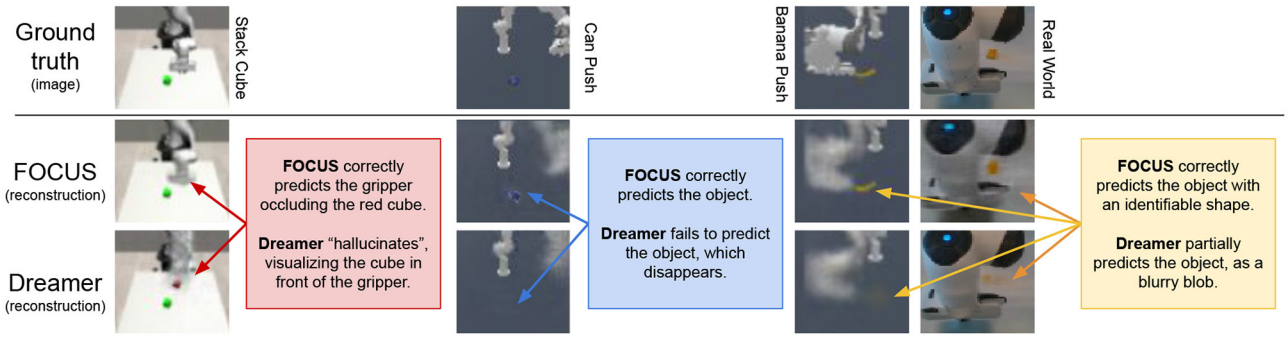


FIGURE 5
Visualizing reconstructions. Comparing reconstructions details of FOCUS's object-centric model and DreamerV2's world model on different environments (RG cube, Master Chef can, Banana, Real World cube). Images and reconstructions are provided with the same resolution as in the models, which is 64x64.

whether the improved object prediction performance is enough to generally improve performance in these tasks, independently of the exploration performance. We consider three baselines for these tasks: *Dreamer*, with the same set of observations provided to FOCUS, *Dreamer (w/ object pos)*, with additional object position information (x,y,z), and *Multi-CNNs* (Yoon et al., 2023) from the OCRL implementation (Yoon et al., 2023), as a model-free RL baseline using an object-centric representation. MultiCNNs extracts an object-centric representation from single observations (no temporal information), and it uses it to train a model-free PPO agent (Schulman et al., 2017). In Dreamer (w/ object pos), we concatenate object position to the proprioception of the agent. To account for the difference in dimensionality between this low-dimensional vector (proprioception + object position) and the large image matrix (64x64x3), we scale the proprioception loss term by a factor of 100. In Figure 7, we compare the final normalized performance in terms of episode rewards.

We observe that FOCUS obtains the highest median and mean performance. This supports the hypothesis that object-centric representations for world models generally improve RL performance for manipulation. When positional information is provided to Dreamer, this helps to improve performance since it is easier for the system to track objects' positions. Still, FOCUS shows an edge in performance, given the higher amount of implicit information available (e.g. orientation, contact, color, ...). Multi-CNNs struggle compared to the other approaches. We speculate this is linked to the lack of temporal consistency in the representation and to the adoption of a less efficient model-free learning strategy.

5.2.2 Information partitioning

We assess whether FOCUS is correctly partitioning the information about each object into their respective latent while storing no additional information from the other elements in the

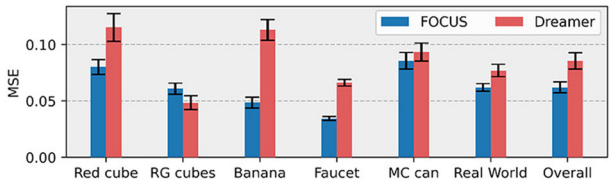


FIGURE 6
Object prediction errors. For each task, MSE is computed over the masked RGB image. Masking is obtained by dilatation (3 pixels) of the object's segmentation mask. Results are averaged over the 10 evaluation episodes.

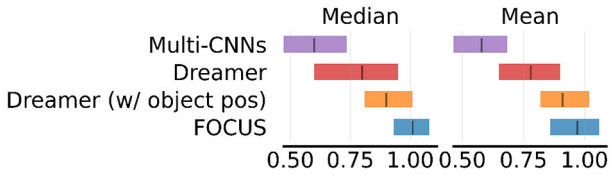


FIGURE 7
Dense reward performance. Results of dense reward experiments across 6 tasks. Experiments are run for 2M steps with three seeds per task and aggregated using RLiable (Agarwal et al., 2021).

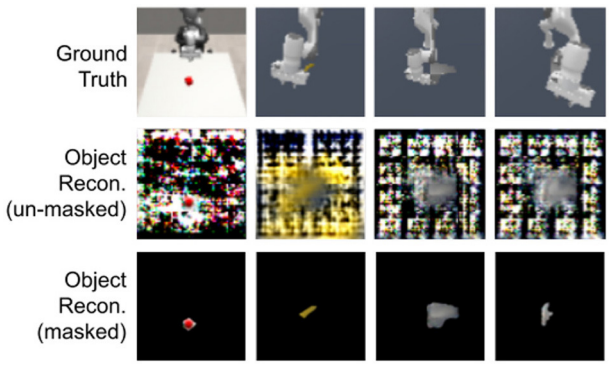
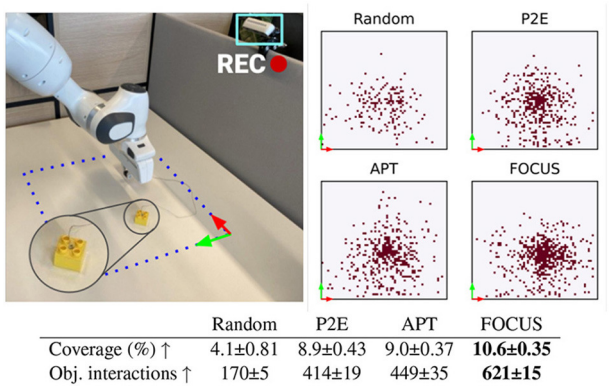


FIGURE 8
Object reconstructions. Unmasked and Masked objects reconstructions of FOCUS. Environments considered are Red cube, Banana, and Faucet. Images and reconstructions are provided with the same resolution as in the models, which is 64 × 64.



	Random	P2E	APT	FOCUS
Coverage (%) ↑	4.1±0.81	8.9±0.43	9.0±0.37	**10.6±0.35**
Obj. interactions ↑	170±5	414±19	449±35	**621±15**

FIGURE 9
Real-world exploration. **Top-Left**: Real-world setup with delimited workspace. **Top-Right**: Positions reached by the cube during the finetuning phase for each method. **Bottom**: number of interactions between the manipulator and the cube, over 3 training seeds.

scene. To get a glimpse into the information separation of FOCUS, we decode the information from the object's latent and we report some examples in Figure 8.

It is evident that the object latent is storing visual information about the object, capturing only a small amount of information from the rest of the image. The "leaked" information is present mostly in the area surrounding the object and we believe is due to the segmentation masks' quality. In the last two columns, we also show examples of occlusion behaviors (partial and full occlusion) by the robotic arm. Despite not seeing the object fully, FOCUS disentangles the object information from the robotic arm and can reconstruct the full unmasked object from the occluded views.

5.3 Real-world object-centric world model

We deploy FOCUS on a Franka Emika robot arm setup. The main issue in the real world comes from the absence of segmentation masks. Similarly to how we did for the MetaWorld experiments, we can adopt the fastSAM model (Kirillov et al., 2023; Zhao et al., 2023) to obtain segmentation masks, given a text prompt (Cheng et al., 2023).

To evaluate the performance of the object-centric world model in a real-world setting, we designed a simple environment featuring a yellow brick placed on a tabletop, as shown in Figure 9. The cube, attached to the robot's end-effector by a string, serves as the primary interactive element. Each episode lasts for 100 steps, after which the robot resets to a designated position, bringing the cube back to approximately the center of the workspace. The robotic arm operates within a constrained 2D plane, indicated by the blue dotted line in Figure 9, with its end-effector height fixed above the tabletop. The robot's gripper remains closed and is not controllable, enabling it to interact with the cube exclusively through pushing movements. The restrictions imposed are for safety reasons due to the nature of exploration, but also to reduce the action space and therefore the amount of data collection required to model the environment.

In order to warm up the training of the world model, we pre-train all agents using a dataset of observations collected by using random actions (50 k interactions, approx 24 h of robot time). We use this dataset to pre-train each world model and exploration strategy for 500 k training steps (both the world model and the policy are updated at every step).

In Figure 6 (second to last histogram), we compare the object reconstruction error of Dreamer and FOCUS for the real-world scenario after pre-training. In general, the implicit segmentation knowledge makes it for more dynamically consistent reconstructions when compared to Dreamer. The latter can

sporadically present artifacts (as depicted in the last column of Figure 5) in the reconstruction, especially for trajectories where there is interaction between the objects and the manipulator.

5.3.1 Exploration evaluation

To evaluate the exploration capabilities of FOCUS in a real-world robotic setting, we fine-tune the pre-trained model for real-time exploration on a robotic arm. The finetuning process spans 10k steps, with each episode consisting of 100 steps. We compare FOCUS against the same exploration baselines proposed during the simulation experimentation, thus P2E, APT, and Random. Results are shown in Figure 9. To confirm what was seen during the simulation experiments, FOCUS has the highest interaction score with the object. The performance gap in terms of interaction between FOCUS and the other baselines is smaller compared to the simulated experiments due to the more simplistic setup adopted for the real-world scenario. The distribution of the object's position achieved during the fine-tuning phase is shown in the top-right part of Figure 9. FOCUS has the highest coverage of positions in the workspace, with the highest concentration around the center of the workspace.

6 Discussion

We presented FOCUS, an object-centric model-based agent that eagerly discovers interactions with objects, enabling one to learn manipulation tasks more efficiently. In our evaluation, we found that not only FOCUS enable solving more sparse reward tasks, but also that the object-centric representation generally improves objects' prediction and manipulation performance.

6.1 Limitations

In our exploration experiments we interact with the environment for 2M steps. All methods require first learning an adequate world model for the explorative agent to be able to robustly imagine which action is going to give the maximum explorative outcome. Indeed, FOCUS starts to show an edge over the other methods after 500 K explorative steps. This consistent amount of training steps makes it challenging to have a full deployment in a complex real-world environment. Nonetheless, exploration approaches can be applied in real-world setups, by simplifying the environment drastically, e.g. restricting the action space (Pathak et al., 2019) or employing high-level actions (Mazzaglia et al., 2024).

The primary limitation of FOCUS is its scalability when applied to scenes containing multiple objects of interest, e.g., more than 2. Since the model depends on segmentation masks to isolate the information for each object, each object reconstruction requires an additional output map, both for the segmentation weights and the RGB channels. This approach results in a larger computational and memory footprint that, despite providing higher performance, is less scalable. For future work, it would be interesting to investigate methods to isolate object information that use more compute-efficient representations, such as deep latent particles (Daniel and Tamar, 2022; Haramati et al., 2024), doing so would retain the benefits of the object-centric approach, while relaxing the computational requirements.

Data availability statement

Publicly available datasets were analyzed in this study. This data can be found here: https://github.com/StefanoFerraro/FOCUS.

Author contributions

SF: Conceptualization, Data curation, Investigation, Methodology, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. PM: Conceptualization, Investigation, Methodology, Project administration, Software, Supervision, Visualization, Writing – original draft, Writing – review & editing. TV: Conceptualization, Supervision, Writing – review & editing. BD: Supervision, Writing – review & editing.

Funding

The author(s) declare that financial support was received for the research and/or publication of this article. This research received funding from the Flemish Government (AI Research Program). Pietro Mazzaglia was funded by a Ph.D. grant from the Flanders Research Foundation (FWO).

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Generative AI statement

The author(s) declare that no Gen AI was used in the creation of this manuscript.

Publisher's note

Supplementary material

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fnbot.2025.1585386/full#supplementary-material

References

Agarwal, R., Schwarzer, M., Castro, P. S., Courville, A., and Bellemare, M. G. (2021). Deep reinforcement learning at the edge of the statistical precipice. *Adv Neural Inf Process Syst*. 34, 29304–29320.

Amodei, D., Olah, C., Steinhardt, J., Christiano, P., Schulman, J., and Man, D. (2016). Concrete problems in AI safety. *arXiv [Preprint]*. arXiv:1606.06565.

Burda, Y., Edwards, H., Pathak, D., Storkey, A., Darrell, T., and Efros, A. A. (2018a). Large-scale study of curiosity-driven learning. *arXiv [Preprint]*. arXiv:1808.04355.

Burda, Y., Edwards, H., Storkey, A., and Klimov, O. (2018b). Exploration by random network distillation. *arXiv [Preprint]*. arXiv:1810.12894.

Burgess, C. P., Matthey, L., Watters, N., Kabra, R., Higgins, I., Botvinick, M., et al. (2019). Monet: unsupervised scene decomposition and representation. *arXiv [Preprint]*. arXiv:1901.11390.

Cheng, Y., Li, L., Xu, Y., Li, X., Yang, Z., Wang, W., et al. (2023). Segment and track anything. *arXiv [Preprint]*. arXiv:2305.06558. doi: 10.48550/arXiv.2305.06558

Clark, J., and Amodei, D. (2016). *Faulty Reward Functions in the Wild*. Available online at: https://openai.com/blog/faulty-reward-functions/ (accessed 19 Februaury, 2022).

Daniel, T., and Tamar, A. (2022). "Unsupervised image representation learning with deep latent particles," in *Proceedings of the 39th International Conference on Machine Learning* (New York: PMLR), 4644–4665.

Deng, F., Jang, I., and Ahn, S. (2022). "Dreamerpro: reconstruction-free model-based reinforcement learning with prototypical representations," in *International Conference on Machine Learning*, 4956–4975.

Dittadi, A., Papa, S., Vita, M. D., Schölkopf, B., Winther, O., and Locatello, F. (2021). Generalization and robustness implications in object-centric learning. *arXiv [Preprint]*. arXiv:2107.00637.

Diuk, C., Cohen, A., and Littman, M. L. (2008). "An object-oriented representation for efficient reinforcement learning," in *Proceedings of the 25th International Conference on Machine Learning*, 240–247.

Ferraro, S., Van de Maele, T., Mazzaglia, P., Verbelen, T., and Dhoedt, B. (2022a). Computational optimization of image-based reinforcement learning for robotics. *Sensors* 22:7382. doi: 10.3390/s22197382

Ferraro, S., Van de Maele, T., Mazzaglia, P., Verbelen, T., and Dhoedt, B. (2022b). Disentangling shape and pose for object-centric deep active inference models. *arXiv [Preprint]*. arXiv:2209.09097. doi: 10.1007/978-3-031-28719-0_3

Ferraro, S., Van de Maele, T., Verbelen, T., and Dhoedt, B. (2023). Symmetry and complexity in object-centric deep active inference models. *Interf. Focus* 13:20220077. doi: 10.1098/rsfs.2022.0077

Fujimoto, S., van Hoof, H., and Meger, D. (2018). "Addressing function approximation error in actor-critic methods," in *International Conference on Machine Learning*, 1587–1596.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. Cambridge, MA: MIT Press. Available online at: http://www.deeplearningbook.org (accessed April 10, 2025).

Greff, K., Kaufman, R. L., Kabra, R., Watters, N., Burgess, C., Zoran, D., et al. (2020). Multi-object representation learning with iterative variational inference. *arXiv [Preprint]*. arXiv:1903.00450.

Gu, J., Xiang, F., Li, X., Ling, Z., Liu, X., Mu, T., et al. (2023). Maniskill2: a unified benchmark for generalizable manipulation skills. *arXiv [Preprint]*. arXiv:2302.04659.

Ha, D., and Schmidhuber, J. (2018). World models. *Zenodo*. doi: 10.5281/ZENODO.1207631

Haarnoja, T., Zhou, A., Abbeel, P., and Levine, S. (2018). Soft actor-critic: off-policy maximum entropy deep reinforcement learning with a stochastic actor. *arXiv [Preprint]*. arXiv:1801.01290.

Hafner, D., Lillicrap, T., Fischer, I., Villegas, R., Ha, D., Lee, H., et al. (2019). "Learning latent dynamics for planning from pixels," in *ICML*, 2555–2565.

Hafner, D., Lillicrap, T. P., Norouzi, M., and Ba, J. (2021). "Mastering atari with discrete world models," in *ICLR*.

Hafner, D., Pasukonis, J., Ba, J., and Lillicrap, T. (2023). Mastering diverse domains through world models. *arXiv [Preprint]*. arXiv:2301.04104. doi: 10.48550/arXiv.2301.04104

Hansen, N., Wang, X., and Su, H. (2022). Temporal difference learning for model predictive control. *arXiv [Preprint]*. arXiv:2203.04955.

Haramati, D., Daniel, T., and Tamar, A. (2024). Entity-centric reinforcement learning for object manipulation from pixels. *arXiv [Preprint]*. arXiv:2404.01220.

Hawkins, J., Ahmad, S., and Cui, Y. (2017). A theory of how columns in the neocortex enable learning the structure of the world. *Front. Neural Circuits* 11:81. doi: 10.3389/fncir.2017.00081

Janner, M., Fu, J., Zhang, M., and Levine, S. (2021). When to trust your model: model-based policy optimization. *arXiv [Preprint]*. arXiv:1906.08253.

Janner, M., Levine, S., Freeman, W. T., Tenenbaum, J. B., Finn, C., and Wu, J. (2019). "Reasoning about physical interactions with object-centric models," in *International Conference on Learning Representations*.

Kalashnikov, D., Irpan, A., Pastor, P., Ibarz, J., Herzog, A., Jang, E., et al. (2018). Qt-Opt: Scalable deep reinforcement learning for vision-based robotic manipulation. *arXiv [Preprint]*. arXiv:1806.10293. doi: 10.48550/Arxiv.1806.10293

Kipf, T., van der Pol, E., and Welling, M. (2020). Contrastive learning of structured world models. *arXiv [Preprint]*. arXiv:1911.12247.

Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., et al. (2023). Segment anything. *arXiv [Preprint]*. arXiv:2304.02643.

Krakovna, V. (2020). Specification Gaming: the Flip Side of AI Ingenuity. Available online at: https://www.deepmind.com/blog/specification-gaming-the-flip-side-of-ai-ingenuity (accessed April 19, 2022).

Lee, A. X., Devin, C., Zhou, Y., Lampe, T., Bousmalis, K., Springenberg, J. T., et al. (2021). Beyond pick-and-place: Tackling robotic stacking of diverse shapes. *arXiv [Preprint]*. arXiv:2110.06192. doi: 10.48550/arXiv.2110.06192

Levine, S., Finn, C., Darrell, T., and Abbeel, P. (2016). End-to-end training of deep visuomotor policies. *J. Mach. Learn. Res.* 17, 1334–1373.

Liu, H., and Abbeel, P. (2021). "Behavior from the void: Unsupervised active pre-training," in Advances in Neural Information Processing Systems, eds. M. Ranzato, A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan (New York: Curran Associates, Inc), 18459–18473.

Locatello, F., Weissenborn, D., Unterthiner, T., Mahendran, A., Heigold, G., Uszkoreit, J., et al. (2020). Object-centric learning with slot attention. *arXiv [Preprint]*. arXiv:2006.15055.

Lu, Y., Hausman, K., Chebotar, Y., Yan, M., Jang, E., Herzog, A., et al. (2021). "AW-Opt: learning robotic skills with imitation andreinforcement at scale," in *5th Annual Conference on Robot Learning (CoRL)*.

Mazzaglia, P., Catal, O., Verbelen, T., and Dhoedt, B. (2022). Curiosity-driven exploration via latent Bayesian surprise. *arXiv [Preprint]*. arXiv:2104.07495.

Mazzaglia, P., Cohen, T., and Dijkman, D. (2024). "Information-driven affordance discovery for efficient robotic manipulation," in *2024 IEEE International Conference on Robotics and Automation (ICRA)* (Yokohama: IEEE).

Mutti, M., Pratissoli, L., and Restelli, M. (2021). Task-agnostic exploration via policy gradient of a non-parametric state entropy estimate. *arXiv [Preprint]*. arXiv:2007.04640.

Nakano, A., Suzuki, M., and Matsuo, Y. (2023). "Interaction-based disentanglement of entities for object-centric world models," in *The Eleventh International Conference on Learning Representations*.

OpenAI, Akkaya, I., Andrychowicz, M., Chociej, M., Litwin, M., McGrew, B., et al. (2019). Solving rubik's cube with a robot hand. *arXiv [Preprint]*. arXiv:1910.07113. doi: 10.48550/arXiv.1910.07113

Oudeyer, P., Kaplan, F., and Hafner, V. V. (2007). Intrinsic motivation systems for autonomous mental development. *IEEE Trans. Evol. Comp.* 11, 265–286. doi: 10.1109/TEVC.2006.890271

Parr, T., Sajid, N., Da Costa, L., Mirza, M. B., and Friston, K. J. (2021). Generative models for active vision. *Front. Neurorobot.* 15:651432. doi: 10.3389/fnbot.2021.651432

Pathak, D., Agrawal, P., Efros, A. A., and Darrell, T. (2017). Curiosity-driven exploration by self-supervised prediction. *arXiv [Preprint]*. arXiv:1705.05363.

Pathak, D., Gandhi, D., and Gupta, A. (2019). Self-supervised exploration via disagreement. *arXiv [Preprint]*. arXiv:1906.04161.

Popov, I., Heess, N., Lillicrap, T., Hafner, R., Barth-Maron, G., Vecerik, M., et al. (2017). Data-efficient deep reinforcement learning for dexterous manipulation. *arXiv [Preprint]*. arXiv:1704.03073.

Rajeswar, S., Ibrahim, C., Surya, N., Golemo, F., Vazquez, D., Courville, A., et al. (2021). Touch-based curiosity for sparse-reward tasks. *arXiv [Preprint]*. arXiv:2104.00442.

Rajeswar, S., Mazzaglia, P., Verbelen, T., Piché, A., Dhoedt, B., Courville, A., et al. (2023). *Mastering the Unsupervised Reinforcement Learning Benchmark From Pixels*.

Sancaktar, C., Blaes, S., and Martius, G. (2022). "Curious exploration via structured world models yields zero-shot object manipulation," in *Advances in Neural Information Processing Systems*, eds. A. H. Oh, A. Agarwal, D. Belgrave, and K. Cho. Available online at: https://openreview.net/forum?id=NnuYZ1el24C

Schmidhuber, J. (1991). "Curious model-building control systems," in *Proceedings 1991 IEEE International Joint Conference on Neural Networks* (Singapore: IEEE), 1458–1463.

Schrittwieser, J., Antonoglou, I., Hubert, T., Simonyan, K., Sifre, L., Schmitt, S., et al. (2020). Mastering atari, go, chess and shogi by planning with a learned model. *Nature* 588, 604–609. doi: 10.1038/s41586-020-03051-4

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., and Klimov, O. (2017). Proximal policy optimization algorithms. *arXiv [Preprint]*. arXiv:1707.06347.

Sekar, R., Rybkin, O., Daniilidis, K., Abbeel, P., Hafner, D., and Pathak, D. (2020). "Planning to explore via self-supervised world models," in *ICML*.

Seo, Y., Chen, L., Shin, J., Lee, H., Abbeel, P., and Lee, K. (2021). State entropy maximization with random encoders for efficient exploration. *arXiv [Preprint]*. arXiv:2102.09430.

Seo, Y., Hafner, D., Liu, H., Liu, F., James, S., Lee, K., et al. (2022). Masked world models for visual control. *arXiv [Preprint]*. arXiv:2206.14244.

Singh, H., Misra, N., Hnizdo, V., Fedorowicz, A., and Demchuk, E. (2003). Nearest neighbor estimates of entropy. *Am. J. Mathem. Managem. Sci.* 23, 301–321. doi: 10.1080/01966324.2003.10737616

Slone, L., Smith, L., and Yu, C. (2019). Self' generated variability in object images predicts vocabulary growth. *Dev. Sci.* 22:12816. doi: 10.1111/desc.12816

Smith, L. B., Jayaraman, S., Clerkin, E., and Yu, C. (2018). The developing infant creates a curriculum for statistical learning. *Trends in Cogn. Sci.* 22, 325–336. doi: 10.1016/j.tics.2018.02.004

Song, Y., Sun, P., Liu, H., Li, Z., Song, W., Xiao, Y., et al. (2024). Scene-driven multimodal knowledge graph construction for embodied ai. *IEEE Trans. Knowl. Data Eng.* 36, 6962–6976. doi: 10.1109/TKDE.2024.3399746

Wu, P., Escontrela, A., Hafner, D., Goldberg, K., and Abbeel, P. (2022). Daydreamer: world models for physical robot learning. *arXiv [Preprint]*. arXiv:2206.14176.

Yang, J., Gao, M., Li, Z., Gao, S., Wang, F., and Zheng, F. (2023). Track anything: segment anything meets videos. *arXiv [Preprint]*. arXiv:2304.11968.

Yoon, J., Wu, Y.-F., Bae, H., and Ahn, S. (2023). An investigation into pre-training object-centric representations for reinforcement learning. *arXiv [Preprint]*. arXiv:2302.04419.

Yu, T., Quillen, D., He, Z., Julian, R., Hausman, K., Finn, C., et al. (2019). "Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning," in *Conference on Robot Learning (CoRL)*. Available online at: https://arxiv.org/abs/1910.10897

Zhao, X., Ding, W., An, Y., Du, Y., Yu, T., Li, M., et al. (2023). Fast segment anything. *arXiv [Preprint]*. arXiv:2306.12156.

Zhu, Y., Wong, J., Mandlekar, A., Martín-Martín, R., Joshi, A., Nasiriany, S., et al. (2020). robosuite: a modular simulation framework and benchmark for robot learning. *arXiv [Preprint]*. arXiv:2009.12293. doi: 10.48550/arXiv.2009.12293