Optimizing Object-based Perception and Control by Free-Energy Principle

Minne Li, Pranav Nashikkar and Jun Wang
University College London
minne.li@cs.ucl.ac.uk, pranav.nashikkar.15@ucl.ac.uk, jun.wang@cs.ucl.ac.uk

Abstract

One of the well-known formulations of human perception is a hierarchical inference model based on the interaction between conceptual knowledge and sensory stimuli from the partially observable environment. This model helps human to learn inductive biases and guides their behaviors by minimizing their surprise of observations. However, most model-based reinforcement learning still lacks the support of object-based physical reasoning. In this paper, we propose Object-based Perception Control (OPC). It combines the learning of perceiving objects from the scene and that of control of the objects in the perceived environments by the free-energy principle. Extensive experiments on high-dimensional pixel environments show that OPC outperforms several strong baselines in accumulated rewards and the quality of perceptual grouping.

1 Introduction

Reinforcement learning (RL), which learns how to map observations to actions to maximize a numerical reward signal, is considered to be the closest form of learning that humans and other animals do [Rescora et al., 1972; Barto et al., 1983]. However, many of the current advances in RL lie in the model-free paradigm and assume the environment state to be fully observable [Mnih et al., 2016]. These group of RL algorithms usually require large amounts of experience and trial-and-error to explore until learning a good policy and ignore the fact that the environment model of many real-world problems is typically unknown. Nevertheless, humans behave in a partially observable world: in order to operate within the constantly changing environment by keeping homeostasis [Kauffman, 1993], humans manage to minimize their surprise of the observation outcome by maintaining and updating a good environment model [Gregory, 1980; Friston, 2005], which enables human to learn new concepts from seeing just a few examples efficiently. Many model-based reinforcement learning (MBRL) approaches have been proposed to perform under partially observable Markov decision processes (POMDPs) and learn an abstract yet informative model of the world [Kaelbling et al., 1998], which helps to reduce the dimension of the input and the amount of training data [Watter et al., 2015; Levine et al., 2016; Ha and Schmidhuber, 2018; Gu et al., 2016; Igl et al., 2018].

However, existing MBRL methods often fail to facilitate common-sense physical reasoning [Battaglia et al., 2013], which learns the structured properties of the environment and provides inductive biases (the prior knowledge of observed objects’ locations, shapes, etc.) to help human make inferences that go beyond the observation [Lake et al., 2017]. Humans learn inductive biases with the interactive environment feedback throughout their lifecycle [Spelke et al., 1992], leading to a unified hierarchical and behavioral-correlated perception model to perceive events and objects from the environment [Lee and Mumford, 2003]. To create AI capable of simulating this human-like inductive-biases learning, we shall build a model to create knowledge through execution-time optimization, rather than a model merely generating static products from offline training [Dehaene et al., 2017].

In this paper, we propose Object-based Perception Control (OPC), a perception model in the context of RL to infer inductive biases from POMDPs with raw pixel observations. We build the object-based inference by segmenting the sensory input into multiple conceptual objects upon unsupervised perceptual grouping [van Steenkiste et al., 2018]. To emphasize the behavioral correlates of perception updating, we impose the trial-and-error on the perceptual grouping process. The perception model is typically hierarchical as stated by the Bayesian brain hypothesis [Dayan et al., 1995; Knill and Pouget, 2004], and is usually formulated as the minimization of free-energy [Neal and Hinton, 1999], which is a close approximation of the surprise. The coherent framework of perception and control becomes beneficial as the perception model helps decision-making with inductive biases, while the decision-making module provides the temporal-difference error [Sutton, 1988] from the interaction with the environment to bias the perception learning towards semantically relevant perceptual representations. Experiments on the Pixel Waterworld environment show that OPC outperforms several strong baselines in terms of accumulated rewards, and the quality and consistency of the perceptual grouping.

2 Preliminary on Bayesian Brain Hypothesis

Many of the classical hierarchical perception models on vision sensory inputs include the Restricted Boltzmann Ma-
chines (RBM) [Hinton, 2012], the predictive coding [Mc-
Clelland and Rumelhart, 1981], and the Helmholtz Ma-
chine [Dayan et al., 1995]. They share a general principle
of Bayesian brain hypothesis, which formulates human per-
ception as a hierarchical inference process based on the
interaction between sensory stimuli (a bottom-up recognition
model \( q(s) \)) and conceptual knowledge (a top-down gen-
erative model \( p(o,s) \)) [Bruner and Goodman, 1947]. The perception then comes as the recognition model formaliz-
ing beliefs about the cause of observations, i.e., to infer and
maximize the posterior \( p(s|o) \) (the probability of different
hidden states given the observation) by inverting the like-
hood model \( p(o|s) \) (the probability of observation given their
causes). The inversion requires maximizing the model evi-
dence \( p(o) \) (the prediction ability of the generative model regard-
ing the observation \( o \)), which is equivalent to minimizing the
surprise

\[
-\log p(o) = -\int q(s) \frac{p(o,s)}{q(s)} ds \\
\leq \int q(s) \left( \frac{q(s)}{p(o,s)} \right) ds = KL(q(s) \| p(s|o)) - \log p(o),
\]

where \( q(s) \) is the bottom-up recognition model. Eq. (1) is
called as the free energy [Neal and Hinton, 1999], which is an
upper bound of the surprise of observations as the KL term
in Eq. (2) is non-negative. The bound becomes tight when the
KL term equals to zero, i.e., the difference between the recogni-
tion model and the posterior probability is minimized.

3 Object-based Perception Control
3.1 Environment Setting
We define the environment as a partially observable Markov
Decision Process (POMDP) represented by tuple \( \Gamma = \langle S, P, A, O, U, R \rangle \), where \( S, A, O \) are the state space, the
action space, and the observation space, respectively. Given two
sets \( X \) and \( Y \), we use \( X \times Y \) to denote the Cartesian product
of \( X \) and \( Y \), i.e., \( X \times Y = \{ (x,y) | x \in X, y \in Y \} \).

For an agent performing in this environment, we consider
its received observation \( o^t \in O \equiv \mathbb{R}^D \) at time step \( t \) as a
visual image (a matrix of pixels) composed of \( K \) objects,
where each pixel \( o_i \) is determined by exactly one object.
At each time step \( t \), we denote as \( s^t \in S \equiv [0,1]^{D \times K} \) the
latent state which encodes the true pixel assignments,
such that \( s^t_{i,k} = 1 \) iff pixel \( o_i^t \) was generated by component \( k \).

\[\sum_k s^t_{i,k} = 1\] the observation \( o^t \) is generated by the
environment following the conditional observation distribution
\( \mathcal{U}_\theta(o^t | s^t) : S \rightarrow O \). Concretely, each pixel \( o_i^t \) is rendered by
one of the object representations \( \theta^t_1, \ldots, \theta^t_K \in \mathbb{R}^{M} \), which
are transformed by a differentiable non-linear function \( f_\theta \) into variables \( \psi^t_k = f_\theta(\theta^t_k) \) for separate pixel-wise distri-
butions \( \mathcal{U}_{\psi^t_k}(o_i^t | s^t_{i,k}) = 1 \). Whether an observation is prob-
able (or not surprised) given an agent’s current state and ac-
tion is quantified by its observed reward \( \log \mathcal{P}(o^t | s^t, a^t) = r^t \in \mathbb{R} \) provided by the environment according to the reward
function \( R(r^t | s^t, a^t) : S \times A \rightarrow \mathbb{R} \). When the environment receives an action \( a^t \in A \), it moves to a new state \( s^{t+1} \) fol-
lowing the transition function \( \mathcal{P}(s^{t+1} | s^t, a^t) : S \times A \rightarrow S \).
In the following sections, we also use \( \rho^t \) to represent the state-
transition tuple \( (s^t, a^t) \) at time step \( t \).

3.2 Learning Object-based Perception Control
To formalize the belief about the way the observation \( o^t \) is
caused, the agent maintains a perception model \( q(s^t) \)
to approximate the probability of different latent states
\( \mathcal{P}(s^t | o^t \preceq t, a^t) \). Using the encoded sufficient statistic of the
history, agent’s inferred belief about the latent state could
serve as the input to its decision-making module (the policy)
\( \pi \). The prediction ability of the perception model at time step
\( t \) is given by the model evidence as

\[\beta^t_\pi,\theta = \prod_{j=t}^{\infty} \mathcal{P}(o^j | \pi^t) = \mathbb{E}_{\rho^t \sim \pi(\rho^t)} \left[ \mathcal{L}(o^t \preceq t | \rho^t) \right]. \] (3)

See details of derivation in the supplementary material. The
goal of the agent is to learn a perception model with high
prediction ability by minimizing the average surprise

\[J_{\pi,\theta} = \sum_{t=0}^{\infty} -\log \beta^t_{\pi,\theta} = \mathbb{E}_{\tau \sim \pi} \left[ -\log \mathcal{L}(o^t \preceq t | \rho^t) \right]. \] (4)

over trajectories \( \tau = (s^0, a^1, s^1, a^1, \ldots) = (\rho^0, \rho^1, \ldots) \)
duced by agent’s policy \( \pi \). We denote the distribution over
initial state as \( \mathcal{U}^{t} : S \rightarrow [0,1] \).

---

1We consider \( \mathcal{U}_{\psi^t_k}(o_i^t | s^t_{i,k}) = 1 \sim \mathcal{N}(o_i^t ; \mu = \psi^t_k, \sigma^2) \).
and the differentiable non-linear function $f_{\theta}$ using a convolutional encoder-decoder architecture with a recurrent neural network as bottleneck, which linearly combine the output of the encoder with $\theta_{l}$ from the previous timestep. The training process is organized in an unsupervised manner by minimizing $\mathbb{E}_{\tau \sim \pi} \left[ -\mathcal{Q}_{\theta}^{t}(\theta^{t+1}) \right]$.

### 3.4 Convergence of the Perception Model Update

Under main assumptions and lemmas as introduced below, we demonstrate the convergence of a sequence of average surprise values $\{J_{\pi, \theta'}\}$ generated by the perception update. The proof is presented by showing that the learning process follows the Global Convergence Theorem [Zangwill, 1969].

**Assumption 1.** $\Omega_{0} = \{\theta \in \Omega : J_{\pi, \theta} \leq J_{\pi, \theta'}\}$ is compact for any $J_{\pi, \theta'} < \infty$.

**Assumption 2.** $J$ is continuous in $\Omega$ and differentiable in the interior of $\Omega$.

The above assumptions lead to the fact that $\{J_{\pi, \theta'}\}$ is bounded for any $\theta' \in \Omega$.

**Lemma 1.** Let $\Omega_{S}$ be the set of stationary points in the interior of $\Omega$, then the mapping $\arg \max_{\theta \in \Omega_{S}} \mathcal{Q}_{\theta}^{t}(\theta^{t+1})$ from Eq. [8] is closed over $\Omega \cap \Omega_{S}$ (the complement of $\Omega_{S}$).

**Proof.** See [Wu, 1983]. A sufficient condition is that $\mathcal{Q}_{\theta}^{t}(\theta^{t+1})$ is continuous in both $\theta^{t+1}$ and $\theta^{t}$.

**Proposition 1.** Let $\Omega_{S}$ be the set of stationary points in the interior of $\Omega$, then (i) $\forall \theta' \in \Omega_{S}$, $J_{\pi, \theta'} \leq J_{\pi, \theta}$ and (ii) $\forall \theta' \in \Omega \cap \Omega_{S}$, $J_{\pi, \theta'} < J_{\pi, \theta}$.

**Proof.** Note that (i) holds true given the condition. To prove (ii), consider any $\theta' \in \Omega \cap \Omega_{S}$, we have

$$
\frac{\partial \mathcal{Q}_{\theta}^{t}(\theta^{t+1})}{\partial \theta^{t+1}} \bigg|_{\theta^{t+1} = \theta^*} = \frac{\partial J_{\pi, \theta}}{\partial \theta^{t+1}} \bigg|_{\theta^{t+1} = \theta^*} \neq 0.
$$

Hence $\mathcal{Q}_{\theta}^{t}(\theta^{t+1})$ is not maximized at $\theta^{t+1} = \theta^*$. Given the perception update described by Eq. [3], we therefore have $\mathcal{Q}_{\theta}^{t}(\theta^{t+1}) > \mathcal{Q}_{\theta}^{t}(\theta^*)$, which implies $J_{\pi, \theta'} < J_{\pi, \theta}$.

**Theorem 1.** Let $\{\theta^t\}$ be a sequence generated by the mapping from Eq. [8] $\Omega_{S}$ be the set of stationary points in the interior of $\Omega$. If Assumptions 1 & 2 Lemma 1 and Proposition 2 are met, then all the limit points of $\{\theta^t\}$ are stationary points (local minima) and $J_{\pi, \theta^*}$ converges monotonically to $J^* = J_{\pi, \theta^*}$ for some stationary point $\theta^*$ in $\Omega_{S}$.

**Proof.** Suppose that $\theta^*$ is a limit point of the sequence $\{\theta^t\}$. Given Assumptions 1 & 2 and Proposition 2, we have that the sequence $\{\theta^t\}$ are contained in a compact set $\Omega_{K} \subset \Omega$. Thus, there is a subsequence $\{\theta^{l}\}$ of $\{\theta^t\}$ such that $\theta^{l} \to \theta^*$ as $l \to \infty$ and $l \in L$.

We first show that $J_{\pi, \theta^*} \to J_{\pi, \theta^*}$ as $l \to \infty$. Given $J$ is continuous in $\Omega$ (Assumption 2), we have $J_{\pi, \theta^*} \to J_{\pi, \theta^*}$ as $l \to \infty$ and $l \in L$, which means

$$\forall \epsilon > 0, \exists \delta (\epsilon) \in L \text{s.t.} \forall l \geq \delta (\epsilon), l \in L, J_{\pi, \theta_l} - J_{\pi, \theta^*} < \epsilon. \tag{10}$$

Given Proposition 2 and Eq. (5), $J$ is therefore monotonically decreasing on the sequence $\{\theta^{l}\}_{l = 0}^{\infty}$, which gives

$$\forall t, J_{\pi, \theta^*} - J_{\pi, \theta^*} \geq 0. \tag{11}$$
Algorithm 1 Learning the Object-based Perception Control

Initialize $\theta$, $\eta$, $\phi$, $\zeta$, $\tau_C$, $T_{epsi}, \tau_o$

Create $N_v$ environments that will execute in parallel while training not finished do

Initialize the history $D_0, D_1, D_2$ with environment rollouts for $t_{\tau_o} + 1$ time-steps under current policy $\pi$

for $T = 1$ to $T_{epsi} - t_{\tau_o}$ do

$d\phi \leftarrow 0$

Get $\eta^{-1}$ from $D_0$

for $t = T$ to $T + t_{\tau_o}$ do

Get $a^t$, $o^t$ from $D_0$, $D_v$ respectively

Feed $\eta_k^{-1} \circ (\psi_k^{-1} - o^t)$ into each of the K RNN copy to get $\theta_k^t$ and forward-output $\psi_k^t$

Compute $\eta_k$ by Eq. 7

$d\phi \leftarrow d\phi + \frac{\partial \left(-Q_{\theta}(\theta^{t+1})\right)}{\partial \phi}$ by Eq. 8

end for

Perform $a^{T+t_{\tau_o}}$ according to policy $\pi_{\pi}(a^{T+t_{\tau_o}}|\theta^{T+t_{\tau_o}})$

Receive reward $r^{T+t_{\tau_o}}$ and new observation $o^{T+t_{\tau_o}+1}$

Store $a^{T+t_{\tau_o}}$, $o^{T+t_{\tau_o}+1}$, $\eta^t$ in $D_{\tau_o}, D_v, D_v$ respectively

Feed $\eta_k^{T+t_{\tau_o}} \circ (\psi_k^{T+t_{\tau_o}} - o^{T+t_{\tau_o}+1})$ into each of the K RNN copy to get $\theta_k$

$y \leftarrow \gamma V_{\zeta}(\theta^{T+t_{\tau_o}+1})$

$d\zeta \leftarrow \nabla \log \pi_{\pi}(a^{T+t_{\tau_o}}|\theta^{T+t_{\tau_o}})(y - V_{\zeta}(\theta^{T+t_{\tau_o}}))$

$d\phi \leftarrow d\phi + \frac{\partial \left(y - V_{\zeta}(\theta^{T+t_{\tau_o}})\right)^2}{\partial \phi}$

end for

Perform synchronous update of $\phi$ using $d\phi$, of $\zeta$ using $d\zeta_v$, and of $\zeta_v$ using $d\zeta_v$

end while

Given Eq. 10, for any $t \geq l(\epsilon)$, we have

$J_{\pi, \theta} - J_{\pi, \theta^*} = J_{\pi, \theta^*} - J_{\pi, \theta^*} + J_{\pi, \theta^*} - J_{\pi, \theta^*} < \epsilon$. (12)

Given Eq. 11 and Eq. 12, we therefore have $J_{\pi, \theta^*} \rightarrow J_{\pi, \theta^*}$ as $t \rightarrow \infty$. We then prove that the limit point $\theta^*$ is a stationary point. Suppose $\theta^*$ is not a stationary point, i.e., $\theta^* \in \Omega \setminus \Omega_S$. We consider the sub-sequence $\{\theta^{t+1}\}_{t \in L}$, which are also contained in the compact set $\Omega_K$. Thus, there is a subsequence $\{\theta^{t+1}\}_{t \in L'}$ of $\{\theta^{t+1}\}_{t \in L}$ such that $\theta^{t+1} \rightarrow \theta^*$ as $t \rightarrow \infty$. On the other hand, since the mapping from Eq. 6 is closed over $\Omega \setminus \Omega_S$ (Lemma 1), and $\theta^* \in \Omega \setminus \Omega_S$, we therefore have $\theta^* \in \arg \max_{\theta^*} Q_{\theta^*}(\theta^{t+1})$, yielding $J_{\pi, \theta^*} < J_{\pi, \theta^*}$ (Proposition 1), which contradicts Eq. 13.

3.5 Decision-making Module Update

The second term in Eq. 5 is the negative of the total reward along trajectory $\tau$, which is straightforward to understand as an agent with a better perception model would receive a higher total reward by conditioning its policy on its inferred belief about the latent state. Intuitively, the additional knowledge provided by the decomposition of high-dimensional observations with respect to disentangled and structured object abstractions could ease the burden of decision-making.

To maximize the total reward along trajectory $\tau$, we follow the conventional Temporal-Difference learning approach [Sutton, 1988] by passing the object abstractions to a small multilayer perceptron (MLP) [Rumelhart et al., 1986] to produce an $(\dim_{\mathcal{A}}(A) + 1)$-dimensional vector, which is split into a $(\dim_{\mathcal{A}}(A))$-dimensional vector of $\pi_{\pi}$’s (the ‘actor’) logits, and a baseline scalar $V_{\zeta}$ (the ‘critic’). The $\pi_{\pi}$ logits are normalized using a softmax function, and used as multinomial distribution from which an action is sampled. The $V_{\zeta}$ is an estimate of the state-value function at the current state, which is given by the last hidden state $\theta$ of the $t_{\tau_o}$-step RNN rollout. On training the decision-making module, the $V_{\zeta}$ is used to compute the temporal-difference error given by

$L_{TD} = (y^{t+1} - V_{\zeta}(\theta^{t+1}))^2, \quad y^{t+1} = r^{t+1} + \gamma V_{\zeta}(\theta^{t+2}),$ (14)

where $\gamma \in [0, 1)$ is a discount factor. $L_{TD}$ is used both to optimize $\pi_{\pi}$ to generate actions with larger total rewards than $V_{\zeta}$ predicts by updating $\zeta$ with respect to the policy gradient $\nabla \zeta \log \pi_{\pi}(a^{t+1}|\theta^{t+1}) (y - V_{\zeta}(\theta^{t+1}))$ and to optimize $V_{\zeta}$ to more accurately estimate state values by updating $\zeta$. By differentiating $L_{TD}$ with respect to $\phi$ enables the gradient-based optimizers to update the perception model. We provide the pseudo-code for one-step TD-learning of the proposed model in Algorithm 1. By grouping objects with respect to the reward, our model could support distinguishing objects with high visual similarities but different meanings, thus helping to accelerate the learning procedure and better understand the environment.

4 Related Work

Model-based deep reinforcement learning algorithms have been shown to be more effective than model-free alternatives in certain tasks [Watter et al., 2015; Levine et al., 2016; Gu et al., 2016; Igl et al., 2018]. However, these models typically produce entangled latent representations for pixel observations, making them unable to facilitate physical reasoning and learn inductive biases. Although [Zambaldi et al., 2018] have used relational mechanism to discover and reason about relevant entities, their model needs additional supervision to label entities with their location information. The method most closely related to our approach is the World Model [Ha and Schmidhuber, 2018], which consists of separately trained models for visual, memorizing, and control purposes, thus preventing the formerly trained components from the guidance provided by latter components. Different from these approaches, OPC provides the decision-making process with object-based abstractions of high-dimensional observations, which naturally contain object-based position information. Also, the unified model breaks down the raw pixels into objects with respect to the reward, our model could support distinguishing objects with high visual similarities but different meanings, thus helping to accelerate the learning procedure and better understand the environment.
models normally assume prior knowledge of object-based representations rather than extracting from high-dimensional observation space [Diuk et al., 2008]. When objects are extracted through learning methods, these models usually require supervised modeling of the object definition, by either comparing the activation spectrum generated from neural network filters with existing types [Garnelo et al., 2016] or leveraging the bounding boxes generated by standard object detection algorithms in computer vision [Keramatia et al., 2018]. On the other hand, OPC follows an unsupervised manner to extract object abstractions by minimizing the free energy of unsupervised perceptual grouping. By biasing the discovered object representations through the TD error, the decision-making process can also benefit from the inductive bias.

5 Experiments

5.1 Pixel Waterworld

We demonstrate the mutual facilitation of object-based perception and reinforcement learning by applying OPC on a modified Waterworld environment [Karpathy, 2015], where the observations are 84 * 84 grayscale raw pixel images composed of an agent and two types of bouncing targets: the poison and the food, as illustrated in Fig. (2). The agent can control its velocity by choosing from four available actions: to apply thrusters to the left, right, up and down. The environment dynamics will then integrate the velocity of the agent to change its position. A negative reward will be awarded to the agent if it touches any poison target, while a positive reward for making contact with any food target. The optimal strategy depends on the number, the moving speed, and the size of objects, thus requiring the agent to infer the underlying physical dynamics of the environment within a given amount of observations. The intuition of this environment is to test the agent performing in the environment find an optimal reward view and using the same mix of n-step returns to update both the $\pi_\zeta$ and the $V_\zeta$. We build the environment with two poison objects and one food object moving in high speed and set the size of the agent 1.5 times smaller than the target. Results are reported with separate runs of three random seeds. Note that the learning procedure of WM is off-line, and requires many more observation samples than OPC as it consists of independent training for each model. Following the early-stopping criteria in [Ha and Schmidhuber, 2018], we report that WM requires 300 times more observation samples to give the results presented in Fig. (3).

Fig. (3a) shows the result of accumulated rewards after each agent has experienced the same amount of observations in its life-cycle, where the agent with OPC achieves the best performance as having the highest accumulated reward than agents with any other models performing in the environment. We believe this advantage owes to the help of inductive biases learned by the perception model (compared to entangled representations extracted by the CNN used in standard A2C), and the joint learning of perception and control of OPC instead of the separate learning of WM. To illustrate agent’s learning process through time, we also present the period reward, which is the accumulated reward during a given period of environment interactions (2e4 of experienced observations) along the agent’s life-cycle. As illustrated in Fig. (3b), OPC significantly improves the sample efficiency of A2C, making the agent performing in the environment find an optimal strategy much quicker than the A2C agent, as well as agents with WM and the random policy. We also find that the standard version of A2C with four parallel running threads gives roughly the same result as the single-thread version of A2C (the same as the decision-making module of OPC), eliminating the potential drawback of single-thread learning.

5.2 Optimize RL through Inductive Biases: Accumulated Reward Comparison

To verify inductive biases facilitating reinforcement learning, we compare OPC against a set of baseline algorithms, including (1) the standard A2C, which uses convolutional layers to transform the raw pixel observations to low dimensional vectors as input for the same MLP described in Sect. [3.5], (2) the World Model [Ha and Schmidhuber, 2018] (WM), a state-of-the-art model-based approach, which learns separate models for visual, memorizing, and control purposes respectively; and (3) the random policy. For both the baseline A2C and the decision-making module of OPC, we follow the convention of [Mnih et al., 2016] by running the algorithm in the forward view and using the same mix of n-step returns to update both the $\pi_\zeta$ and the $V_\zeta$. We build the environment with two poison objects and one food object moving in high speed and set the size of the agent 1.5 times smaller than the target. Results are reported with separate runs of three random seeds. Note that the learning procedure of WM is off-line, and requires many more observation samples than OPC as it consists of independent training for each model. Following the early-stopping criteria in [Ha and Schmidhuber, 2018], we report that WM requires 300 times more observation samples to give the results presented in Fig. (3).

Fig. (3a) shows the result of accumulated rewards after each agent has experienced the same amount of observations in its life-cycle, where the agent with OPC achieves the best performance as having the highest accumulated reward than agents with any other models performing in the environment. We believe this advantage owes to the help of inductive biases learned by the perception model (compared to entangled representations extracted by the CNN used in standard A2C), and the joint learning of perception and control of OPC instead of the separate learning of WM. To illustrate agent’s learning process through time, we also present the period reward, which is the accumulated reward during a given period of environment interactions (2e4 of experienced observations) along the agent’s life-cycle. As illustrated in Fig. (3b), OPC significantly improves the sample efficiency of A2C, making the agent performing in the environment find an optimal strategy much quicker than the A2C agent, as well as agents with WM and the random policy. We also find that the standard version of A2C with four parallel running threads gives roughly the same result as the single-thread version of A2C (the same as the decision-making module of OPC), eliminating the potential drawback of single-thread learning.

5.3 Optimize Perception Update through the TD Signal: Perceptual Grouping Results

To demonstrate the facilitation of perception update by reinforcement learning, we compare OPC against a perception model, namely OP, with the same architecture but no guid-
−1.5
−1.0
−0.5
0.0
3rd Rewards (×1e3)

Figure 4: The soft-assignment $\eta$ from neighboring time steps, visualized by coloring each pixel $i$ according to their distribution over classes $\eta_i$. $\alpha$ is the target observation. The agent location difference between models owes to separately learned policies.

5.4 Effectiveness of the Hyperparameter Setting

We further investigate the influence of hyperparameters towards the learning ability of OPC, by changing the number of recurrent copies $K$, the rollout steps for recurrent iteration $t_{ro}$, and the use of relational mechanism across object representations $\theta$ described in [van Steenkiste et al., 2018], and present the results of the period reward across different hyperparameter settings in Fig. 5.

As illustrated in Fig. 5a, the number of recurrent copies $K$ affects the stability of OPC learning, as OPC with $K = 3$ has experienced larger variance during the agent’s life-cycle. We believe the difference comes from the environment dynamics as we have visually four objects in the environment. During the earlier stage of interacting with the environment, OPC tries to group each object into a distinct class; thus a different number of $K$ against the number of objects in the environment will confuse the perception model and lead to unstable learning. Although different $K$ settings might affect the learning stability and slow down the convergence, OPC can still find an optimal strategy within a given amount of observations, thus being superior to other baseline models. Meanwhile, the use of relational mechanism has limited impact on OPC, possibly because OPC doesn’t rely on the prediction ability of the perception model, but instead benefits from the learned inductive biases.

In Fig. 5b, we compare OPC with different steps of recurrent rollout. A shorter rollout means fewer rounds of perception update, thus suffering from slower convergence in terms of the number of experienced observations. However, choosing shorter rollout steps can still be beneficial for simpler environments, as the required rounds of perception update will be smaller, thus configuring with shorter rollout steps can find the optimal strategy quicker in terms of wall training time.

6 Conclusions

In this paper, we propose Object-based Perception Control (OPC), demonstrating the mutual facilitation of hierarchical object-based perception and reinforcement learning under the Bayesian brain hypothesis derived from the free-energy principle. Extensive experiments on high-dimensional pixel environments show that OPC outperforms several strong baseline policies, indicating the mutual facilitation of hierarchical object-based perception and reinforcement learning. In future work, we intend to investigate OPC with more types of inductive biases and test the model performance in a wider variety of environments.

References

[Barto et al., 1983] A. G. Barto, R. S. Sutton, and C. W. Anderson. Neuronlike adaptive elements that can solve difficult learning control problems. IEEE transactions on Systems, Man, & Cybernetics, 13(5):834–846, 1983.

[Battaglia et al., 2013] Peter W Battaglia, Jessica B Hamrick, and Joshua B Tenenbaum. Simulation as an engine of physical scene understanding. Proceedings of the National Academy of Sciences, page 201306572, 2013.

[Bengio, 2013] Yoshua Bengio. Deep learning of representations: Looking forward. In International Conference on Statistical Language and Speech Processing, pages 1–37. Springer, 2013.
[Bruner and Goodman, 1947] Jerome S Bruner and Cecile C Good-
man. Value and need as organizing factors in perception. The journal of abnormal and social psychology, 42(1):33, 1947.

[Dayan et al., 1995] Peter Dayan, Geoffrey E Hinton, Radford M Neal, and Richard S Zemel. The helmholtz machine. Neural computation, 7(5):889–904, 1995.

[Dehaene et al., 2017] Stanislas Dehaene, Hakwan Lau, and Sid Koiri. What is consciousness, and could machines have it? Science, 358(6362):486–492, 2017.

[Diuk et al., 2008] Carlos Diuk, Andre Cohen, and Michael L. Littman. An object-oriented representation for efficient reinforcement learning. In Proceedings of the 25th International Conference on Machine Learning, ICML ’08, pages 240–247, New York, NY, USA, 2008. ACM.

[Friston, 2010] Karl Friston. The free-energy principle: a unified computational account. Philosophical Transactions of the Royal Society of London B: Biological Sciences, 365(1456):815–836, 2010.

[Friston, 2010] Karl Friston. The free-energy principle: a unified brain theory? Nature Reviews Neuroscience, 11:127 EP –, 01 2010.

[Gallup, 1970] Gordon G. Gallup. Chimpanzees: Self-recognition. Science, 167(3914):86–87, 1970.

[Garnelo et al., 2016] Marta Garnelo, Kai Arulkumaran, and Murray Shanahan. Towards deep symbolic reinforcement learning. arXiv preprint arXiv:1609.0531x, 2016.

[Gregory, 1980] Richard Langton Gregory. Perceptions as hypotheses. Phil. Trans. R. Soc. Lond. B, 290(1038):181–197, 1980.

[Gu et al., 2016] Shixiang Gu, Timothy Lillicrap, Ilya Sutskever, and Sergey Levine. Deep q-learning with model-based acceleration. In Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48, ICML’16, pages 2829–2838. JMLR.org, 2016.

[Ha and Schmidhuber, 2018] David Ha and Jürgen Schmidhuber. Recurrent world models facilitate policy evolution. In Advances in Neural Information Processing Systems 31, pages 2451–2463. Curran Associates, Inc., 2018. https://worldmodels.github.io

[Hinton, 2012] Geoffrey E. Hinton. A Practical Guide to Training Restricted Boltzmann Machines, pages 599–619. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.

[Igl et al., 2018] Maximilian Igl, Luisa Zintgraf, Tuan Anh Le, Frank Wood, and Shimon Whiteson. Deep reinforcement learning for POMDPs. In Proceedings of the 35th International Conference on Machine Learning, volume 80 of Proceedings of Machine Learning Research, pages 2117–2126. Stockholmsmässan, Stockholm Sweden, 10–15 Jul 2018. PMLR.

[Kaelbling et al., 1998] Leslie Pack秦aelbling, Michael L. Littman, and Richard S. Sutton. Planning and acting in partially observable stochastic domains. Artificial intelligence, 101(1-2):99–134, 1998.

[Karpathy, 2015] Andrej Karpathy. Reinforcejs: Waterworld. https://cs.stanford.edu/people/karpathy/reinforcejs/waterworld.html. 2015. Accessed: 2018-11-20.

[Kauffman, 1993] Stuart A Kauffman. The origins of order: Self-organization and selection in evolution. OUP USA, 1993.

[Keramati et al., 2018] Ramtin Keramati, Jay Whang, Patrick Cho, and Emma Brunskill. Strategic object oriented reinforcement learning. arXiv preprint arXiv:1806.00175, 2018.

[Knill and Pouget, 2004] David C Knill and Alexandre Pouget. The bayesian brain: the role of uncertainty in neural coding and computation. TRENDS in Neurosciences, 27(12):712–719, 2004.

[Lake et al., 2017] Brenden M Lake, Tomer D Ullman, Joshua B Tenenbaum, and Samuel J Gershman. Building machines that learn and think like people. Behavioral and Brain Sciences, 40, 2017.

[Lee and Mumford, 2003] Tai Sing Lee and David Mumford. Hierarchical bayesian inference in the visual cortex. JOSA A, 20(7):1434–1448, 2003.

[Levine et al., 2016] Sergey Levine, Chelsea Finn, Trevor Darrell, and Pieter Abbeel. End-to-end training of deep visuomotor policies. The Journal of Machine Learning Research, 17(1):1334–1373, 2016.

McClelland, 1981 James L McClelland and David E Rumelhart. An interactive activation model of context effects in letter perception: I. an account of basic findings. Psychological review, 88(5):375, 1981.

[Mnih et al., 2016] Volodymyr Mnih, Adria P. Badia, Mehdi Mirza, Alex Graves, Tim Lillicrap, David Silver, and Koray Kavukcuoglu. Asynchronous methods for deep reinforcement learning. In Proceedings of The 33rd International Conference on Machine Learning, volume 48 of Proceedings of Machine Learning Research, pages 1928–1937, New York, New York, USA, 20–22 Jun 2016. PMLR.

[Neal and Hinton, 1999] Radford M. Neal and Geoffrey E. Hinton. Learning in graphical models. chapter A View of the EM Algorithm That Justifies Incremental, Sparse, and Other Variants, pages 355–368. MIT Press, Cambridge, MA, USA, 1999.

[Rescorla et al., 1972] Robert A Rescorla, Allan R Wagner, et al. A theory of pavlovian conditioning: Variations in the effectiveness of reinforcement and nonreinforcement. Classical conditioning II: Current research and theory, 2:64–99, 1972.

[Rumelhart et al., 1986] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Parallel distributed processing: Explorations in the microstructure of cognition, vol. 1. chapter Learning Internal Representations by Error Propagation, pages 318–362. MIT Press, Cambridge, MA, USA, 1986.

[Schmidhuber, 1992] Jürgen Schmidhuber. Learning factorial codes by predictability minimization. Neural Computation, 4(6):863–879, 1992.

[Speleka et al., 1992] Elizabeth S Spelke, Karen Breinlinger, Janet Macomber, and Kristen Jacobson. Origins of knowledge. Psychological review, 99(4):605, 1992.

[Sutton, 1988] Richard S. Sutton. Learning to predict by the method of temporal differences. Machine Learning, 3(1):9–44, Aug 1988.

[van Steenkiste et al., 2018] Sjoerd van Steenkiste, Michael Chang, Klaus Greff, and Jürgen Schmidhuber. Relational neural expectation maximization: Unsupervised discovery of objects and their interactions. In International Conference on Learning Representations, 2018.

[Watter et al., 2015] Manuel Watter, Jost Springenberg, Joschka Boedecker, and Martin Riedmiller. Embed to control: A locally linear latent dynamics model for control from raw images. In Advances in neural information processing systems, pages 2746–2754, 2015.

[Wu, 1983] CF Jeff Wu. On the convergence properties of the em algorithm. The Annals of statistics, pages 95–103, 1983.

[Zambaldi et al., 2018] Vinícius Flores Zambaldi, David Raposo, Adam Santoro, Victor Bapst, Yujia Li, Igor Babuschkin, Karl Tuyls, David P. Reichert, Timothy P. Lillicrap, Edward Lockhart, Murray Shanahan, Victoria Langston, Razvan Pascanu, Matthew Botvinick, Oriol Vinyals, and Peter W. Battaglia. Relational deep reinforcement learning. CoRR, abs/1806.01830, 2018.

[Zangwill, 1969] Willard I Zangwill. Nonlinear programming: a unified approach, volume 196. Prentice-Hall Englewood Cliffs, NJ, 1969.
A  Detailed Review on Bayesian Brain Hypothesis

Many of the classical hierarchical perception models on vision sensory inputs include the Helmholtz Machine [Dayan et al., 1995], the predictive coding [McClelland and Rumelhart, 1981; Rao and Ballard, 1999; Clark, 2013], the Restricted Boltzmann Machines (RBMs) [Hinton, 2012], and the Variational Auto-Encoders (VAEs) [Kingma and Welling, 2013]. Although they vary from the architecture design, the training procedure, or the optimization goals, they share a general principle that the brain is constantly generating and amending hypotheses of sensory input at varying levels of abstraction. This principle comes from the Bayesian brain hypothesis, which derives from the idea of analysis-by-synthesis [Neisser, 2014] and uses Bayesian probability theory to formulate human perception as a hierarchical inference process based on the interaction between sensory stimuli (a bottom-up recognition model \(q(s)\)) and conceptual knowledge (a top-down generative model \(p(o, s)\)).

The perception model actively predicts and explains its sensations while being updated using sensory inputs [von Helmholtz and Southall, 2005], e.g., raw pixels of high-dimensional environment observations. The generative model is decomposed into a likelihood \(p(o|s)\) (the probability of observation given their causes) and the prior belief \(p(s)\) (the a priori probability of those causes). The perception then comes as the recognition model formalizing beliefs about the way observations are caused, i.e., to infer the posterior \(p(s|o)\) (the probability of different hidden states given the observation) by inverting the likelihood model. This inversion is the same as minimizing the difference between the recognition model and the posterior probability to optimize free energy, as we shall show below.

The inference of \(p(s|o)\) via Bayesian theory requires us to compute and maximize the model evidence \(p(o)\), which quantifies the prediction ability of the generative model regarding the observation \(o\) the better the model, the higher the probability of the observation \(p(o)\). Maximizing model evidence is equivalent to minimizing the surprise \(-\log p(o)\), which follows the primary goal of any intelligent agent operating in a changing environment. However, an intelligent agent cannot directly decide whether an observation is probable, since the evaluation of surprise by direct marginalization \(-\log \int p(o, s)\,ds\) is intractable due to the integration over the hidden variable \(s\). Instead, the agent keeps a recognition model \(q(s)\) and therefore manages to minimize

\[
-\log p(o) = -\log \int_s q(s) \frac{p(o, s)}{q(s)}\,ds
\]

\[
\leq \int_s q(s) \log \left( \frac{q(s)}{p(o, s)} \right)\,ds
= KL(q(s)||p(s|o)) - \log p(o).
\]

The inequality in Eq. (15) stands on the convex property of the logarithm and the Jensen’s inequality, i.e., the convex transformation of a mean is less than or equal to the mean applied after convex transformation [Jensen, 1906]. Eq. (15) is called as the free energy [Neal and Hinton, 1999; Hinton and van Camp, 1993], which is an upper bound of the surprise of observations as the KL term in Eq. (16) is non-negative. The bound becomes tight when the KL term equals to zero, i.e., the difference between the recognition model and the posterior probability is minimized. The minimization of free energy is thus tractable by two distinct ways: (1) change perceptual representations (predictions) by updating the perception model \(q(s)\), so that they approximate the conditional density on the causes of particular observations \(p(s|o)\), and (2) change sensory input (observations) by taking actions on the environment to change its states, so that they conform to expectations [Adams et al., 2013; Friston et al., 2017].

B  Additional Related Work

Human-like computing, which aims at endowing machines with human-like perceptual, reasoning and learning abilities, has drawn considerable attention from the recent progress in artificial intelligence (AI) [Lake, 2014; Lake et al., 2015; Graves et al., 2016; Baker et al., 2017]. Reinforcement learning (RL), which learns how to map observations to actions to maximize a numerical reward signal, is considered to be the closest form of learning that humans and other animals do [Rescorla et al., 1972; Barto et al., 1983; Barto et al., 1989; Sutton and Barto, 1990]. Being capable of reasoning about the motions of inanimate objects even at early ages in infancy [Baillargeon et al., 1983; Spelke, 1990], humans gradually develop their cortical systems to learn physical conceptions with the interactive environment feedback [Spelke et al., 1992], leading to a unified hierarchical and behavioral-correlated perception model to perceive events and objects from the environment [Mumford, 1992; Lee and Mumford, 2003]. Different from model-free methods [Silver et al., 2017], which treats learning as making good predictions by discovering patterns correlated to large rewards directly from the environments, model-based deep reinforcement learning algorithms have been shown to be more effective in certain tasks [Levine and Abbeel, 2014; Watter et al., 2015; Levine et al., 2016; Gu et al., 2016; Finn and Levine, 2017; Igl et al., 2018].

The object-based approach, which recognizes decomposed objects from the environment observations, has attracted considerable attention in the field of reinforcement learning as well [Asai and Fukunaga, 2017; Zhu et al., 2018; Goel et al., 2018]. LatPlan [Asai and Fukunaga, 2017] uses a variational auto-encoder (VAE) to generate a problem representation from raw pixel observations in an unsupervised manner, but requires certain constraint on the environment as objects can only appear in a limited number of discrete locations, e.g., each piece of an image-based instance of the 8-puzzle. OODP [Zhu et al., 2018] is very close related to our work by proposing a complicated compositional architecture combining background extraction,
static object detection, and dynamic object detection. One of the limitations of OODP is that it requires the background to be fixed through time. MOREL \cite{Goel et al., 2018} applies optical flow in video sequences and feeds the learned features to model-free RL frameworks as the position and velocity information of moving objects. Although MOREL ignores the relational reasoning among objects, it could be served as a supplementary module to our proposed method to better benefit from existing advances in computer vision. On the other hand, OPC follows an unsupervised manner to extract object abstractions by minimizing the free energy of unsupervised perceptual grouping \cite{van Steenkiste et al., 2018}. By biasing the discovered object representations through the temporal-difference error, he decision-making process can also benefit from the inductive bias by representing states, actions and policies using the relational language, which facilitates the generalization over goals, states, and actions \cite{Dzeroski et al., 2001}.

In addition to the exchange between the agent and the environment, we also consider the physical interactions between object-pairs from the raw sensory input by equipping the perception model with the relational mechanism, which has been extensively discussed in both physical reasoning \cite{Chang et al., 2016; Battaglia et al., 2016; Santoro et al., 2017} and reinforcement learning literature \cite{Dzeroski et al., 2001; Driessens and Dzeroski, 2004; Zambaldi et al., 2018}.

\section*{C Experiment Details}

\textbf{OPC} In all experiments we trained the perception model using ADAM \cite{Kingma and Ba, 2014} with default parameters and a batch size of 32. Each input consists of a sequence of binary $84 \times 84$ images containing two poison objects (two circles) and one food object (a rectangle) that start in random positions and float within the image for $t_{ro}$ steps. These frames were thresholded at 0.0001 to obtain binary images and added with bit-flip noise ($p = 0.2$). We used a convolutional encoder-decoder architecture inspired by recent GANs \cite{Chen et al., 2016} with a recurrent neural network as bottleneck, where the encoder used the same network architecture from \cite{Mnih et al., 2013} as

\begin{enumerate}
\item 8 $\times$ 8 conv. 16 ELU. stride 4. layer norm
\item 4 $\times$ 4 conv. 32 ELU. stride 2. layer norm
\item fully connected. 256 ELU. layer norm
\item recurrent. 256 Sigmoid. layer norm on the output
\item fully connected. 256 RELU. layer norm
\item fully connected. $10 \times 10 \times 32$ RELU. layer norm
\item 4 $\times$ 4 reshape 2 nearest-neighbour, conv. 16 RELU. layer norm
\item 8 $\times$ 8 reshape 4 nearest-neighbour, conv. 1 Sigmoid
\end{enumerate}

We used the Advantage Actor-Critic (A2C) \cite{Mnih et al., 2016} with an MLP policy as the decision making module of OPC. The MLP policy added a 512-unit fully connected layer with rectifier nonlinearity after layer 4 of the perception model. The decision making module had two sets of outputs – a softmax output with one entry per action representing the probability of selecting the action, and a single linear output representing the value function. The decision making module was trained using RMSProp \cite{Tieleman and Hinton, 2012} with a learning rate of $\gamma = 0.99$, an RMSProp decay factor of 0.99, and performed updates after every 5 actions.

\textbf{A2C} We used the same convolutional architecture as the encoder of the perception model of OPC (layer 1 to 3), followed by a fully connected layer with 512 hidden units followed by a rectifier nonlinearity. The A2C was trained using the same setting as the decision making module of OPC.

\textbf{WM-A2C} We use the same setting as \cite{Ha and Schmidhuber, 2018} to separately train the V model and the M model. The experience is generated off-line by a random policy operating in the Pixel Waterworld environment. We concatenated the output of the V model and the M model as the A2C input, and trained A2C using the same setting as introduced above.

\section*{References}

\cite{Adams et al., 2013} Rick A Adams, Stewart Shipp, and Karl J Friston. Predictions not commands: active inference in the motor system. \textit{Brain Structure and Function}, 218(3):611–643, 2013.

\cite{Asai and Fukunaga, 2017} Masataro Asai and Alex Fukunaga. Classical planning in deep latent space: Bridging the subsymbolic-symbolic boundary, 2017.

\cite{Baillargeon et al., 1985} Renee Baillargeon, Elizabeth S Spelke, and Stanley Wasserman. Object permanence in five-month-old infants. \textit{Cognition}, 20(3):191–208, 1985.

\cite{Baker et al., 2017} Chris L. Baker, Julian Jara-Ettinger, Rebecca Saxe, and Joshua B. Tenenbaum. Rational quantitative attribution of beliefs, desires and percepts in human mentalizing. \textit{Nature Human Behaviour}, 1:0064 EP –, 03 2017.

\cite{Barto et al., 1983} A. G. Barto, R. S. Sutton, and C. W. Anderson. Neuronlike adaptive elements that can solve difficult learning control problems. \textit{IEEE transactions on Systems, Man, & Cybernetics}, 13(5):834–846, 1983.
[Barto et al., 1989] Andrew G Barto, Richard S Sutton, and Christopher JCH Watkins. Learning and sequential decision making. In *Learning and computational neuroscience*. Citeseer, 1989.

[Battaglia et al., 2016] Peter Battaglia, Razvan Pascanu, Matthew Lai, Danilo Jimenez Rezende, et al. Interaction networks for learning about objects, relations and physics. In *Advances in neural information processing systems*, pages 4502–4510, 2016.

[Bruner and Goodman, 1947] Jerome S Bruner and Cecile C Goodman. Value and need as organizing factors in perception. *The journal of abnormal and social psychology_, 42(1):33, 1947.

[Bruner and Postman, 1949] Jerome S Bruner and Leo Postman. On the perception of incongruity: A paradigm. *Journal of personality_, 18(2):206–223, 1949.

[Chang et al., 2016] Michael B Chang, Tomer Ullman, Antonio Torralba, and Joshua B Tenenbaum. A compositional object-based approach to learning physical dynamics. *arXiv preprint arXiv:1612.00341*, 2016.

[Chen et al., 2016] Xi Chen, Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. InfoGAN: Interpretable representation learning by information maximizing generative adversarial nets. In D. D. Lee, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances In Neural Information Processing Systems 29*, pages 2172–2180. Curran Associates, Inc., 2016.

[Clark, 2013] Andy Clark. Whatever next? predictive brains, situated agents, and the future of cognitive science. *Behavioral and brain sciences_, 36(3):181–204, 2013.

[Dayan et al., 1995] Peter Dayan, Geoffrey E Hinton, Radford M Neal, and Richard S Zemel. The helmholtz machine. *Neural computation_, 7(5):889–904, 1995.

[Driessens and Džeroski, 2004] Kurt Driessens and Sašo Džeroski. Integrating guidance into relational reinforcement learning. *Machine Learning*, 57(3):271–304, 2004.

[Džeroski et al., 2001] Sašo Džeroski, Luc De Raedt, and Kurt Driessens. Relational reinforcement learning. *Machine learning_, 43(1-2):7–52, 2001.

[Friston, 2010] Karl Friston. The free-energy principle: a unified brain theory? *Nature Reviews Neuroscience_, 11:127 EP –, 01 2010.

[Goel et al., 2018] Vikash Goel, Jameson Weng, and Pascal Poupart. Unsupervised video object segmentation for deep reinforcement learning. In *Advances in Neural Information Processing Systems 31*, pages 5688–5699. Curran Associates, Inc., 2018.

[Graves et al., 2016] Alex Graves, Greg Wayne, Malcolm Reynolds, Tim Harley, Ivo Danihelka, Agnieszka Grabska-Barwińska, Sergio Gómez Colmenarejo, Edward Grefenstette, Tiago Ramalho, John Agapiou, AdriàPuigdomènec Badia, Karl Moritz Hermann, Yori Zwols, Georg Ostrovski, Adam Cain, Helen King, Christopher Summerfield, Phil Blunsom, Koray Kavukcuoglu, and Demis Hassabis. Hybrid computing using a neural network with dynamic external memory. *Nature_, 538:471 EP –, 10 2016.

[Gu et al., 2016] Shixiang Gu, Timothy Lillicrap, Ilya Sutskever, and Sergey Levine. Continuous deep q-learning with model-based acceleration. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48*, ICML’16, pages 2829–2838. JMLR.org, 2016.

[Ha and Schmidhuber, 2018] David Ha and Jürgen Schmidhuber. Recurrent world models facilitate policy evolution. In *Advances in Neural Information Processing Systems 31*, pages 2451–2463. Curran Associates, Inc., 2018.

[Hinton and van Camp, 1993] Geoffrey E. Hinton and Drew van Camp. Keeping the neural networks simple by minimizing the description length of the weights. In *Proceedings of the Sixth Annual Conference on Computational Learning Theory*, COLT ’93, pages 5–13, New York, NY, USA, 1993. ACM.

[Hinton, 2012] Geoffrey E. Hinton. *A Practical Guide to Training Restricted Boltzmann Machines*, pages 599–619. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.

[Igl et al., 2018] Maximilian Igl, Luisa Zintgraf, Tuan Anh Le, Frank Wood, and Shimon Whiteson. Deep variational reinforcement learning for POMDPs. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 2117–2126, Stockholmsmässan, Stockholm Sweden, 10–15 Jul 2018. PMLR.
[Jensen, 1906] Johan Ludwig William Valdemar Jensen. Sur les fonctions convexes et les inégalités entre les valeurs moyennes. *Acta mathematica*, 30(1):175–193, 1906.

[Kingma and Ba, 2014] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[Kingma and Welling, 2013] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[Lake et al., 2015] Brenden M. Lake, Ruslan Salakhutdinov, and Joshua B. Tenenbaum. Human-level concept learning through probabilistic program induction. *Science*, 350(6266):1332–1338, 2015.

[Lake, 2014] Brenden M Lake. *Towards more human-like concept learning in machines: Compositionality, causality, and learning-to-learn*. PhD thesis, Massachusetts Institute of Technology, 2014.

[Lee and Mumford, 2013] Tai Sing Lee and David Mumford. Hierarchical bayesian inference in the visual cortex. *JOSA A*, 20(7):1434–1448, 2003.

[Levine and Abbeel, 2014] Sergey Levine and Pieter Abbeel. Learning neural network policies with guided policy search under unknown dynamics. In *Advances in Neural Information Processing Systems*, pages 1071–1079, 2014.

[Levine et al., 2016] Sergey Levine, Chelsea Finn, Trevor Darrell, and Pieter Abbeel. End-to-end training of deep visuomotor policies. *The Journal of Machine Learning Research*, 17(1):1334–1373, 2016.

[Mackay, 1956] D. Mackay. The epistemological problem for automata. In *Automata Studies*, pages 235–251. Princeton University Press, Princeton, NJ, 1956.

[McClelland and Rumelhart, 1981] James L McClelland and David E Rumelhart. An interactive activation model of context effects in letter perception: I. an account of basic findings. *Psychological review*, 88(5):375, 1981.

[Mnih et al., 2013] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, 2013.

[Mnih et al., 2016] Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. Asynchronous methods for deep reinforcement learning. In *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 1928–1937, New York, New York, USA, 20–22 Jun 2016. PMLR.

[Mumford, 1992] David Mumford. On the computational architecture of the neocortex. *Biological cybernetics*, 66(3):241–251, 1992.

[Neal and Hinton, 1999] Radford M. Neal and Geoffrey E. Hinton. Learning in graphical models. chapter A View of the EM Algorithm That Justifies Incremental, Sparse, and Other Variants, pages 355–368. MIT Press, Cambridge, MA, USA, 1999.

[Neal, 1992] Radford M Neal. Connectionist learning of belief networks. *Artificial intelligence*, 56(1):71–113, 1992.

[Neisser, 2014] Ulric Neisser. *Cognitive psychology: Classic edition*. Psychology Press, 2014.

[Rao and Ballard, 1999] Rajesh P. N. Rao and Dana H. Ballard. Predictive coding in the visual cortex: a functional interpretation of some extra-classical receptive-field effects. *Nature Neuroscience*, 2:79 EP –, 01 1999.

[Rescorla et al., 1972] Robert A Rescorla, Allan R Wagner, et al. A theory of pavlovian conditioning: Variations in the effectiveness of reinforcement and nonreinforcement. *Classical conditioning II: Current research and theory*, 2:64–99, 1972.

[Santoro et al., 2017] Adam Santoro, David Raposo, David G Barrett, Mateusz Malinowski, Razvan Pascanu, Peter Battaglia, and Timothy Lillicrap. A simple neural network module for relational reasoning. In *Advances in neural information processing systems*, pages 4967–4976, 2017.

[Silver et al., 2017] David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, et al. Mastering the game of go without human knowledge. *Nature*, 550(7676):354, 2017.

[Spelke et al., 1992] Elizabeth S Spelke, Karen Breinlinger, Janet Macomber, and Kristen Jacobson. Origins of knowledge. *Psychological review*, 99(4):605, 1992.

[Spelke, 1990] Elizabeth S. Spelke. Principles of object perception. *Cognitive Science*, 14(1):29 – 56, 1990.

[Sutton and Barto, 1990] R. S. Sutton and A. G. Barto. Time-derivative models of Pavlovian reinforcement. In *Learning and Computational Neuroscience*, pages 497–537. MIT Press, Cambridge, MA, 1990.

[Tieleman and Hinton, 2012] Tijmen Tieleman and Geoffrey Hinton. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural networks for machine learning*, 4(2):26–31, 2012.
[van Steenkiste et al., 2018] Sjoerd van Steenkiste, Michael Chang, Klaus Greff, and Jürgen Schmidhuber. Relational neural expectation maximization: Unsupervised discovery of objects and their interactions. In International Conference on Learning Representations, 2018.

[von Helmholtz and Southall, 2005] Hermann von Helmholtz and James Powell Cocke Southall. Treatise on physiological optics, volume 3. Courier Corporation, 2005.

[Watter et al., 2015] Manuel Watter, Jost Springenberg, Joschka Böedecker, and Martin Riedmiller. Embed to control: A locally linear latent dynamics model for control from raw images. In Advances in neural information processing systems, pages 2746–2754, 2015.

[Zambaldi et al., 2018] Vinícius Flores Zambaldi, David Raposo, Adam Santoro, Victor Bapst, Yujia Li, Igor Babuschkin, Karl Tuyls, David P. Reichert, Timothy P. Lillicrap, Edward Lockhart, Murray Shanahan, Victoria Langston, Razvan Pascanu, Matthew Botvinick, Oriol Vinyals, and Peter W. Battaglia. Relational deep reinforcement learning. CoRR, abs/1806.01830, 2018.

[Zhu et al., 2018] Guangxiang Zhu, Zhiao Huang, and Chongjie Zhang. Object-oriented dynamics predictor. In Advances in Neural Information Processing Systems 31, pages 9826–9837. Curran Associates, Inc., 2018.
D Details of Derivation

D.1 Derivation of Eq. (3)

\[
\beta_t^{t-1} = \prod_{j=t}^{\infty} p_{\theta^t}(o^j) \\
= \prod_{j=t}^{\infty} \sum_{s^t, a^t} p_{\theta^t}(o^j, s^t, a^t) \\
= \prod_{j=t}^{\infty} \sum_{s^t, a^t} p(s^t, a^t) p_{\theta^t}(o^j|s^t, a^t) \\
= \prod_{j=t}^{\infty} \mathbb{E}_{(s^t, a^t) \sim p(s^t, a^t)} [p_{\theta^t}(o^{t+1}|s^t, a^t)] .
\]

D.2 Derivation of Eq. (5)

\[
\mathbb{E}_{\tau \sim \pi} \left[ -\log p_{\theta^t}(o^{t+1}|s^t, a^t) \right] \\
= \mathbb{E}_{\tau \sim \pi} \left[ -\log p_{\theta^t}(o^t|s^t, a^t) \prod_{i=1}^{D} p_{\psi_i} (o_{i}^{t+1} | s^t, a^t) \right] \\
= \mathbb{E}_{\tau \sim \pi} \left[ -\log \prod_{i=1}^{D} p_{\psi_i} (o_{i}^{t+1} | s_{i}^{t+1}, s^t, a^t) \right] + \mathbb{E}_{\tau \sim \pi} \left[ -\log p_{\theta^t}(o^t|s^t, a^t) \right] \\
= \mathbb{E}_{\tau \sim \pi} \left[ -\log \prod_{i=1}^{D} \sum_{s_{i}^{t+1}} q(s_{i}^{t+1} | s^t, a^t) \frac{p_{\psi_i} (o_{i}^{t+1} | s_{i}^{t+1}, s^t, a^t)}{q(s_{i}^{t+1} | s^t, a^t)} \right] + \mathbb{E}_{\tau \sim \pi} \left[ -\log p_{\theta^t}(o^t|s^t, a^t) \right] \\
\leq \mathbb{E}_{\tau \sim \pi} \left[ \sum_{i=1}^{D} \sum_{s_{i}^{t+1}} q(s_{i}^{t+1} | s^t, a^t) \log \frac{q(s_{i}^{t+1} | s^t, a^t)}{p_{\psi_i} (o_{i}^{t+1} | s_{i}^{t+1}, s^t, a^t)} \right] + \mathbb{E}_{\tau \sim \pi} \left[ -\log p_{\theta^t}(o^t|s^t, a^t) \right] \\
= \mathbb{E}_{\tau \sim \pi} \left[ \sum_{i=1}^{D} \left[ KL(q(s_{i}^{t+1} | s^t, a^t) \| p_{\psi_i} (o_{i}^{t+1} | s_{i}^{t+1}, s^t, a^t)) \right] + \mathbb{E}_{\tau \sim \pi} \left[ -\log p_{\theta^t}(o^t|s^t, a^t) \right] + \mathbb{E}_{\tau \sim \pi} \left[ -\log p_{\theta^t}(o^t|s^t, a^t) \right] .
\]

D.3 Derivation of Eq. (8)

\[
\theta^{t+1} = \arg \min_{\theta^{t+1}} \sum_{i=1}^{D} \sum_{s_{i}^{t+1}} q(s_{i}^{t+1} | s^t, a^t) \log \frac{q(s_{i}^{t+1} | s^t, a^t)}{p_{\psi_i^{(t+1)}} (o_{i}^{t+1}, s_{i}^{t+1} | s^t, a^t)} \]
\[
= \arg \min_{\theta^{t+1}} \sum_{i=1}^{D} \sum_{s_{i}^{t+1}} \eta_t^i \log \frac{\eta_t^i}{p_{\psi_i^{(t+1)}} (o_{i}^{t+1}, s_{i}^{t+1} | s^t, a^t)} \text{ drop terms which are constant with respect to } \theta^{t+1} \\
= \arg \min_{\theta^{t+1}} \sum_{i=1}^{D} \sum_{s_{i}^{t+1}} \eta_t^i \log p_{\psi_i^{(t+1)}} (o_{i}^{t+1}, s_{i}^{t+1} | s^t, a^t) \\
= \arg \max_{\theta^{t+1}} \mathbb{E}_{s_{i}^{t+1} \sim \eta_t^i} \left[ \log p_{\psi_i^{(t+1)}} (o_{i}^{t+1}, s_{i}^{t+1} | s^t, a^t) \right] \\
\approx \arg \max_{\theta^{t+1}} Q_{\theta^t} (\theta^{t+1})
\]
D.4 Derivation of Eq. (9)

\[
\frac{\partial Q}{\partial \theta_k^{t+1}} = \frac{\partial Q}{\partial \psi_k^{t+1}} \cdot \frac{\partial (\psi_k^{t+1})^T}{\partial (\theta_k^{t+1})^T} \\
= \left[ \frac{\partial Q}{\partial \psi_1^{t+1}} \cdots \frac{\partial Q}{\partial \psi_D^{t+1}} \right] \cdot \left[ \begin{array}{l} \frac{\partial \psi_1^{t+1}}{\partial \theta_k^{t+1}} \\ \frac{\partial \psi_2^{t+1}}{\partial \theta_k^{t+1}} \\ \vdots \\ \frac{\partial \psi_D^{t+1}}{\partial \theta_k^{t+1}} \end{array} \right] \\
= \sum_{i=1}^D \frac{\partial Q}{\partial \psi_i^{t+1}} \cdot \frac{\partial \psi_i^{t+1}}{\partial \theta_k^{t+1}} \\
= \sum_{i=1}^D \frac{\partial}{\partial \psi_i^{t+1}} \left[ \sum_{s_i^{t+1}} p_{\psi_i^{t+1}}(s_i^{t+1} | o_i^{t+1}, s^t, a^t) \log p_{\psi_i^{t+1}}(o_i^{t+1} | s_i^{t+1}) \cdot p(s_i^{t+1} | s^t, a^t) \right] \cdot \frac{\partial \psi_i^{t+1}}{\partial \theta_k^{t+1}} \\
= \sum_{i=1}^D \frac{\partial}{\partial \psi_i^{t+1}} \left[ \sum_{s_i^{t+1}} p_{\psi_i^{t+1}}(s_i^{t+1} | o_i^{t+1}, s^t, a^t) \log p_{\psi_i^{t+1}}(o_i^{t+1} | s_i^{t+1}) \cdot p(s_i^{t+1} | s^t, a^t) \right] \cdot \frac{\partial \psi_i^{t+1}}{\partial \theta_k^{t+1}} \\
= \sum_{i=1}^D \frac{\partial}{\partial \psi_i^{t+1}} \left[ \sum_{s_i^{t+1}} p_{\psi_i^{t+1}}(s_i^{t+1} | o_i^{t+1}, s^t, a^t) \right] \cdot \frac{\partial \psi_i^{t+1}}{\partial \theta_k^{t+1}} \\
= \sum_{i=1}^D \frac{\partial}{\partial \psi_i^{t+1}} \left[ \frac{1}{2} \log \sigma - \frac{(o_i^{t+1} - \psi_i^{t+1})^2}{2\sigma^2} \right] \cdot \frac{\partial \psi_i^{t+1}}{\partial \theta_k^{t+1}} \\
= \sum_{i=1}^D \frac{\partial}{\partial \psi_i^{t+1}} \left[ \frac{1}{2} \log 2\pi \sigma^2 - \frac{(o_i^{t+1} - \psi_i^{t+1})^2}{2\sigma^2} \right] \cdot \frac{\partial \psi_i^{t+1}}{\partial \theta_k^{t+1}} \\
= \sum_{i=1}^D \frac{\partial}{\partial \psi_i^{t+1}} \left[ \frac{1}{2} \left( \frac{2(o_i^{t+1} - \psi_i^{t+1})}{\sigma^2} \right) \right] \cdot \frac{\partial \psi_i^{t+1}}{\partial \theta_k^{t+1}} \\
= \sum_{i=1}^D \frac{\partial}{\partial \psi_i^{t+1}} \left[ \frac{1}{2} \psi_i^{t+1} - \frac{o_i^{t+1}}{\sigma^2} \right] \cdot \frac{\partial \psi_i^{t+1}}{\partial \theta_k^{t+1}}.