Push, See, Predict: Emergent Perception Through Intrinsically Motivated Play

Orestis Konstantaropoulos[1], Mehdi Khamassi[2], Petros Maragos[3] and George Retsinas[4]

Abstract— **Unlike conventional vision systems that rely on passive observation, biological agents learn through physical interaction. Can a robot similarly develop an understanding of its environment purely through interaction, without prior knowledge or external supervision? In this work, we explore how artificial agents can autonomously learn via intrinsic motivation, much like how children engage in curious free play. We propose a novel, fully self-supervised, object-centric learning framework. The system first segments visual input into discrete entities using Slot Attention, trained on data collected from random robotic actions. A graph-based world model is then trained to predict object-centric dynamics but initially struggles to capture object motion due to the limited diversity of the initial interactions. To overcome this, we introduce an intrinsically motivated reward signal based on world model's prediction error, which drives a policy to collect more informative trajectories. This results in up to three times more object displacement than random actions, significantly enriching the dataset. Fine-tuning both the vision and world model on these data improves prediction and reconstruction performance. We validate our method in a simulated robotic environment with diverse objects, demonstrating that meaningful visual and physical representations can emerge entirely from self-supervised interaction. This highlights the potential of intrinsically motivated, object-centric learning for autonomous world perception and modeling.**

I. INTRODUCTION

Today's state-of-the-art AI models continue to break new ground in computer vision and machine learning [1], [2] advancing rapidly across various domains, including image and video classification, semantic segmentation and decision-making. Despite these impressive achievements, the cognitive abilities and world understanding of animals and humans still surpass those of current machine learning (ML) systems. Unlike humans, who require minimal exposure to new tasks to adapt and succeed, ML systems depend on vast amounts of data along with carefully designed supervisory signals from human experts. These data samples must be independent and identically distributed (i.i.d.); when the domain or data distribution shifts, typical AI models struggle to generalize effectively. Humans, on the other hand, can master new tasks with limited practice and data. For example, children learn new tasks and objects fast by re-exploiting structured knowledge from previous interactions and actively testing predictions [3], [4]. Therefore, it is essential to thoroughly study and draw inspiration from biological cognition and its underlying principles in our attempt to develop more reliable and efficient artificial systems [5].

In this work, we explore active perception through a cognitively inspired approach while leveraging recent advances in machine learning. Our goal is to emulate the behavior of a human infant, who perceives the world and interacts with it to incrementally enhance their understanding of the environment and develop an internal model of it. Unlike traditional AI models that rely heavily on supervisory signals and large datasets, we aim to investigate whether a model can learn in a self-supervised manner within a novel, open-ended environment. To achieve this, we study world model learning while a robotic arm actively explores objects on a table so as to maximize an intrinsically motivated epistemic reward function. Importantly, while to our knowledge most previous developmental robotics approaches to world model learning used a fixed visual module [6], [7], here we use the data generated through active interaction with objects to simultaneously learn the world model and the vision module entirely from scratch, without any external supervision, pretrained modules or external datasets. We first show that our proposed epistemic reward generates actions that lead up to three times more object displacement than random actions. We then show that the resulting policy leads to both world model improvement (i.e., better prediction of state-action-state dynamics) and visual reconstruction.

To validate the proposed methodology, we conduct experiments using a simulated robotic arm in a tabletop environment with a diverse set of objects. Our results demonstrate that the proposed approach is effective in scenarios where no supervision signals, pretrained modules or large datasets are available.

In summary, **our key contributions** are:

1) We adopt an **object-centric approach** and develop a world model capable of predicting the future states of object representations across frames.
2) We **train** both the vision and world models **entirely from scratch**, without any supervision or external datasets.
3) We design an **intrinsically motivated reward** signal based on the world model's prediction error, which effectively filters out noise introduced by imperfect

[1] O. Konstantaropoulos is with the Robotics Institute, Athena Research Center 15125 Maroussi, Greece and the School of ECE, National Technical University of Athens, Greece `o.konstantaropoulos@athenarc.com`

[2] M. Khamassi is with the Institute of Intelligent Systems and Robotics, CNRS, Sorbonne University, Paris, France

[3] P. Maragos is with the RI/Athena RC, 15125 Maroussi, Greece, the School of ECE, NTUA, Athens, Greece and the HERON - Center of Excellence in Robotics, Athens, Greece `petros.maragos@athenarc.gr`

[4] G. Retsinas is with the RI/AthenaRC, 15125 Maroussi, Greece `george.retsinas@athenarc.gr`

This work has received funding from the EU's Horizon Europe research and innovation programme under grant no. 101070381 (project: PILLAR-Robots)

doi: 10.1109/ICDL63968.2025.11204356

models and encourages policies **that result in up to three times more object displacement** on average.

4) We show that **fine-tuning the models** on data collected via the learned policy significantly **improves the robot's world understanding**, measured by prediction and reconstruction performance of both the vision and world models.
5) We validate our approach in a simulated robotic environment, demonstrating clear improvements in interaction quality and object-centric prediction.

II. Related Work

A. Visual segmentation through self-supervised learning

This work builds upon several research domains at the intersection of computer vision, robot learning, and cognitively inspired machine learning. At its core, our problem can be framed as a computer vision task: detecting and segmenting objects in a scene through self-supervised learning. A growing body of work has focused on object-centric methods for unsupervised video object segmentation, where "objects" are treated as fundamental building blocks of representation.

Most object-centric approaches rely on a reconstruction objective to uncover meaningful structure in visual scenes and can be broadly categorized into scene-mixture and spatial-attention models. Scene-mixture models [8], [9], [10] interpret a scene as a composition of multiple latent components, each reconstructed by a generative model. In contrast, spatial-attention models [11], [12], [13] encode geometric properties of objects, such as location, scale, and presence, by decomposing an image into background and foreground elements, with the foreground further represented as a set of individual object representations. These representations have been shown to support downstream control tasks [14]. Generally, this object-centric paradigm aligns with the causal structure of the physical world [15].

B. Intrinsically-motivated RL

Our approach is also grounded in the principles of intrinsically-motivated reinforcement learning (RL), where agents acquire useful representations or behaviors without externally defined rewards. Instead, intrinsic objectives or self-supervised signals guide learning. For instance, off-policy deep RL has been applied to learn visual grasping strategies from self-supervised data collection [16], [17].

A central element of our method is an intrinsic reward function that promotes exploration and the generation of novel, informative trajectories. Pathak et al. [18] introduced curiosity-driven exploration using prediction error as an intrinsic reward. Related approaches [19], [20] include leveraging ensembles of predictive models to quantify uncertainty and encouraging exploration where model predictions disagree most, or maximizing entropy in the learned state space.

C. Robot active perception

While approaches mentioned above study intrinsic motivations to learn policies or world models relying on a fixed perception module, here we are interested in simulteanously improving perception, which corresponds to an active perception process. Several studies in the literature have explored active perception [21], [22], [23]. More recently, researchers have begun integrating deep learning into this concept. Pinto et al [24] in The Curious Robot, argue that biological agents learn visual representations through physical interactions and build a system that pushes, pokes, grasps and observes objects in a tabletop environment to learn such representations. This system is trained to predict the outcomes of these robotic tasks with data annotated in a self-supervised manner. The extracted representations have been shown to be beneficial for simple downstream control tasks.

Similarly, Pathak et al. [25] propose a self-supervised approach to object segmentation through interaction with the environment. Their agent maintains a segmentation hypothesis, manipulates a hypothesized object through random actions, and updates the model based on the difference between visual frames captured before and after each action. These interactions provide a noisy yet informative signal that enhances the initial segmentation hypothesis over time. In [26], Sancaktar et al. demonstrated that a preliminary phase of curiosity-driven free play can enhance downstream task performance. Their system employs an ensemble of world models to plan actions that maximize epistemic uncertainty. Cobra [27] also adopts a task-free intrinsically motivated exploration approach. Using unsupervised learning, they build object-based transition models of their environment optimizing a pixel-based loss function, which they use in a model-based reinforcement learning setting. Although similar in spirit, [26] relies on proprioceptive state information to train world models from which rewards are derived, while [27] evaluates their approach only in a two-dimensional, visually simplistic environment. *To the best of our knowledge, we are the first to propose a fully self-supervised, object-centric framework that intrinsically enhances an agent's world perception and modeling in a visually complex environment.*

III. Methodology

A. Overview

Our method follows a self-supervised, object-centric pipeline composed of five key stages: a) We begin by collecting an initial dataset of image sequences, generated by an agent performing random actions in the environment. b) We train a self-supervised vision model, based on Slot Attention, to segment scenes into object-like components and learn structured object-centric representations. c) Using the frozen vision model, we train a world model to predict future visual representations conditioned on the agent's actions. This world model is of low quality due to the mainly uninformative actions in the initial dataset. d) We train a policy using an intrinsic reward derived from the prediction error of the world model, encouraging the agent to explore states where the model is uncertain. e) Finally, we use the learned policy to collect more informative trajectories involving object interactions, and fine-tune both the vision and world model on this richer dataset. A summary of the proposed framework is illustrated in Fig. 1.

B. Self-Supervised Vision Encoder

To achieve self-supervised, object-centric vision representations, we employ Slot Attention [28], a state-of-the-art unsupervised method for object discovery and segmentation. It introduces latent variables, referred to as slots, which bind to perceptual inputs via a differentiable attention mechanism, capturing distinct parts of the scene as object-like entities.

The core idea is simple: decompose an image into slots and reconstruct it from them. This self-supervised pipeline comprises two main components, a Vision Encoder and a Slot Decoder, enabling Slot Attention to learn object-centric representations without supervision.

Vision Encoder: Given a video with frames $\mathbf{I}_t$ at timestep t, each frame is encoded into K slots $\mathbf{S}_t \in \mathbb{R}^{K \times D}$, where the number of slots K and the slots' dimensionality D are predefined parameters set by the user. The image is first processed by a DNN backbone, producing a feature map $\mathbf{h}_t = f_{enc}(\mathbf{I}_t) \in \mathbb{R}^{N \times D_f}$, with N spatial locations and feature dimensionality D_f. The Slot Attention module then iteratively binds slots to input features via a differentiable attention mechanism. Slots compete to explain different regions of the scene and are progressively refined through learnable projection layers, an MLP, and a GRU [29]. This process requires initializing slot representations, originally done randomly.

Slot Decoder: Each slot is decoded using a spatial broadcast decoder [30] to reconstruct its corresponding scene region. Each slot $\mathbf{s}_k$ produces a reconstruction $\hat{\mathbf{I}}_k$ and a mask $\mathbf{\Pi}_k$, normalized via spatial softmax. The final image is then formed by combining all components:

$$\hat{\mathbf{I}} = \sum_{k=1}^{K} \mathbf{\Pi}_k \odot \hat{\mathbf{I}}_k. \tag{1}$$

For simplicity, we use $\hat{\mathbf{I}}_t = f_{dec}(\mathbf{S}_t)$, where f_{dec} denotes the entire slot decoder pipeline.

Training through Reconstruction: we can now train, the CNN backbone of the encoder, the trainable parts of the slot attention, as well as the decoder, jointly by simply reconstructing the original input, across all frames:

$$L_{rec} = \sum_{t=1}^{T} \|\hat{\mathbf{I}}_t - \mathbf{I}_t\|^2 \tag{2}$$

Modifications on Slot Attention: While object-centric learning shows promise, most methods rely on weak supervision or large-scale pretrained encoders. We instead train Slot Attention **from scratch** using only data collected autonomously by a robotic arm.

To improve performance, we modify the original pipeline by (i) using a ResNet encoder with a larger receptive field, and (ii) pretraining the encoder as part of an autoencoder on our dataset to improve convergence and stability.

Entropy-based loss term: Optimizing Slot Attention to produce clean, localized masks remains challenging in our setting, thus we introduce an additional loss term that penalizes the entropy of the spatial masks $\mathbf{\Pi}_k$ associated with each slot. This promotes low-entropy, coherent masks, reducing noise and improving segmentation.

From Images to Videos: To capture temporal dynamics, we employ a sequential extension of Slot Attention, designed to operate on videos. Instead of randomly re-initializing the slots for each consecutive input frame, a predictor module serves as a transition function to model temporal relationships, as done in SAVI [31].

C. World Model for Predicting Future Slots

Our goal is to enable models to decompose scenes into objects, infer their properties, and understand inter-object relations. A key step is training a world model that predicts physical dynamics and action outcomes—e.g., what happens when an object is pushed. Given that our system already extracts structured slot representations, building such a model becomes straightforward.

Following Kipf et al. [32], we use a fully connected graph neural network (GNN) as an action-conditioned transition model over slot representations. It learns object-level abstractions from offline tuples $(\mathbf{S}_t, a_t, \mathbf{S}_{t+r})$, where $\mathbf{S}_t$ are the slots at time t, a_t is the action, and $\mathbf{S}_{t+r}$ the resulting slots after a fixed interval r, corresponding to the frame where the action's effect is observed.

Implementation-wise, a node update function f_{node} and an edge update function f_{edge} is shared across all nodes and edges, both implemented as MLPs. A single round of message passing updates is performed using the following equations on individual slots $\mathbf{s}_t^k$: $e_t(i,j) = f_{edge}([\mathbf{s}_t^i, \mathbf{s}_t^j])$ and $\Delta\mathbf{s}_t^j = f_{\text{node}}([\mathbf{s}_t^j, a_t^j, \sum_{i \neq j} e_t(i,j)])$.

Typically, we can train the world model using the MSE loss over the predicted slots:

$$L_{pred} = \|\mathbf{S}_t + T(\mathbf{S}_t, a_t) - \mathbf{S}_{t+r}\|_2 \tag{3}$$

To improve sample efficiency, a contrastive hinge loss is also employed, where the predicted state transition is compared to a randomly corrupted state representation $\mathbf{S}^c$:

$$L_{hinge} = max(0, \gamma - \|\mathbf{S}_t + T(\mathbf{S}_t, a_t) - \mathbf{S}^c\|_2) \tag{4}$$

We also introduce a third reconstruction loss term back on the pixel space employing the frozen vision decoder f_{dec} and ensuring that the predicted slots can be decoded to actual changes in the robot's environment:

$$L_{rec} = \|f_{dec}(\mathbf{S}_t + T(\mathbf{S}_t, a_t)) - \mathbf{I}_{t+r}\|_2 \tag{5}$$

Overall, the loss is defined as $L_{wm} = L_{pred} + L_{hinge} + \alpha L_{rec}$. In practice, we used $\gamma = 10$ and $\alpha = 10^3$.

D. Designing an intrinsically motivated reward

A key component of our approach is an intrinsic reward function that guides exploration by encouraging trajectories with high information gain for both the vision and world models. Following [18], we base the reward on the world model's prediction error. Raw prediction error can be noisy—especially early in training—due to limitations in the

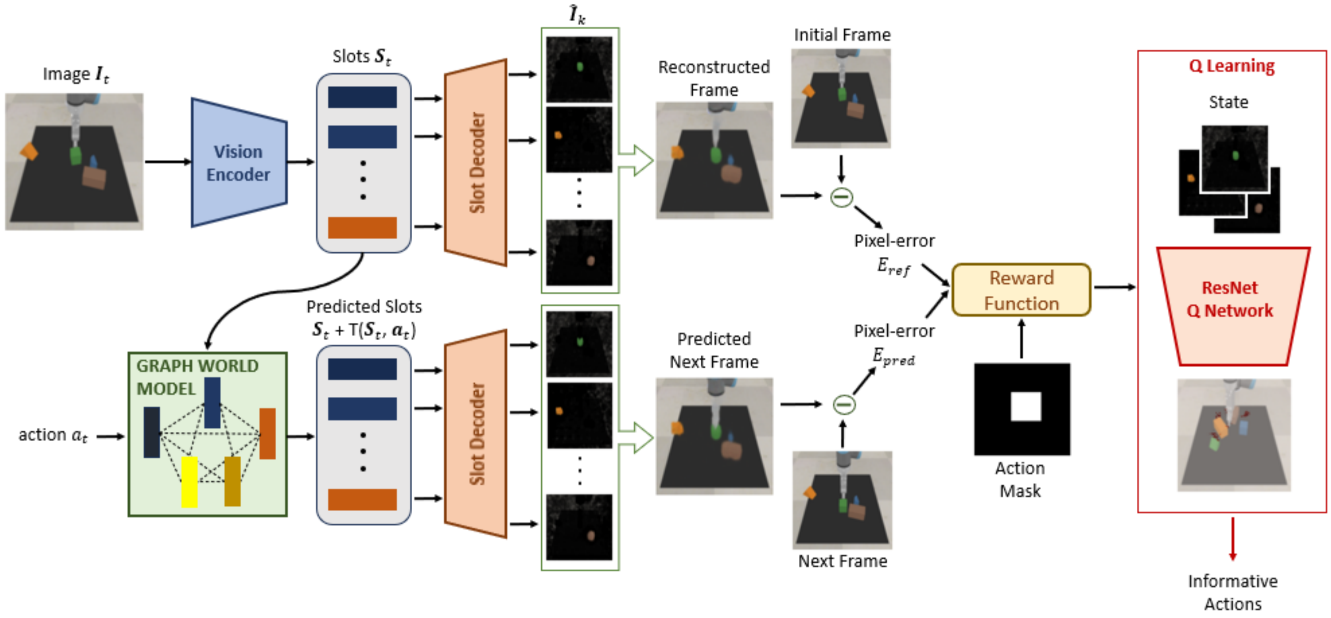

Fig. 1: Overview of our proposed framework: The input image $\mathbf{I}_t$ is processed by the vision encoder to extract K slot representations which can be decoded to the reconstructed image $\hat{\mathbf{I}}_t$. A graph-based world model takes the K slots and the corresponding action as input to predict the future frame $\mathbf{I}_{t+r}$. A reward function is then computed based on the prediction error, which guides a Q-network to propose informative actions.

models. To address this, we compute the error in pixel space as the difference between predicted and actual future frames:

$$\mathbf{E}_{pred} = (f_{dec}(\mathbf{S}_t + T(\mathbf{S}_t, a_t)) - \mathbf{I}_{t+r})^2 \tag{6}$$

However, this error may also capture biases introduced by the decoder, reflecting the limitations of the current vision model rather than true prediction error. To isolate this, we compute a reference error using only the static vision reconstruction pipeline:

$$\mathbf{E}_{ref} = (f_{dec}(\mathbf{S}_t) - \mathbf{I}_t)^2 \tag{7}$$

Only prediction errors exceeding this reference reflect epistemic uncertainty, namely the model's current ignorance about the outcome of its own actions.

To localize the learning signal, the reward is computed only over a region of the image centered on the action, denoted by a spatial mask $\mathbf{M}$. This reduces the influence of irrelevant changes and focuses the reward on action-relevant areas. Overall, the intrinsic reward is calculated as:

$$r_t = \sum_i \sum_j [\mathbf{M} \odot max((\mathbf{E}_{pred} - \mathbf{E}_{ref}), 0)]_{ij} \tag{8}$$

E. Training a policy

In order to train a policy to maximize this reward, we frame the problem as a Markov Decision Process (MDP). States are defined by the K segmentation masks, concatenated in the channel dimension, which are produced by the frozen vision model at each timestep and actions correspond to the robotic arm's movements. We train the policy using Double Q-learning [33], with the Q-network implemented as a ResNet. The network takes as input the segmentation masks and outputs a spatial map indicating the expected value of performing a pushing action at each pixel.

F. Refining the models with informative trajectories

The learned policy generates image sequences that are novel and informative for both the vision and world model. We leverage these new trajectories by fine-tuning the models on them. Unlike the initial dataset, which predominantly featured static scenes or gripper motion, the new data includes rich object interactions. As a result, the models gain exposure to object dynamics, allowing them to improve both *reconstruction quality* and *predictive accuracy*.

IV. The Experimental Setup

To evaluate our proposed methods and hypotheses, we designed an appropriate experimental setup using the CoppeliaSim simulation environment [34]. The experiment involves a UR5 robotic arm interacting with objects placed on a tabletop. These objects are simple three-dimensional geometric shapes with unknown colors and physical properties. The robotic arm executes non-preemptive motion primitives, parameterized by three values: the x and y coordinates on the table and the movement orientation. The orientation is selected from 16 discretized options. The only type of action performed is pushing. To simplify the experimental setup, we restrict our experiments to pushing actions with a fixed orientation. The primary objective of our system is to detect objects in the scene and infer as much as possible about their physical properties. A key challenge in this problem setting is that our models have no access to any supervision signal or prior knowledge. *Instead, all models are trained*

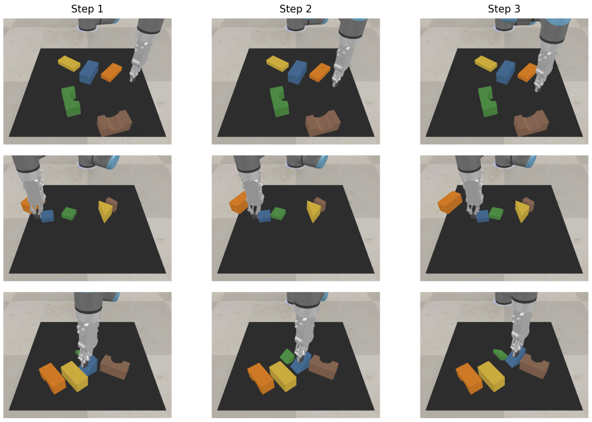

Fig. 2: Experimental Setup: Sequences of robotic arm push actions in our tabletop environment. Top: random push with no object movement. Middle and bottom: heuristic pushes causing object movement.

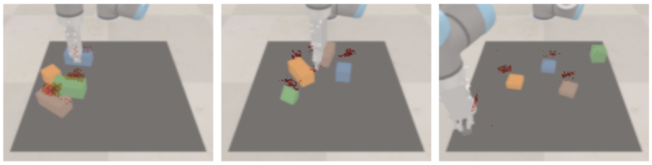

Fig. 3: Policy visualization: We visualize the output of the Q-network by marking as red the pixels that our policy suggests. The red pixels clearly illustrate that the agent has learned to prefer actions that push objects.

from scratch, relying solely on the sequences of interactions experienced by the robotic arm.

Initially, the robotic arm interacts randomly with the environment, selecting from the available pushing actions. During this phase, we construct a dataset comprising tuples of robotic actions and corresponding image sequences. This dataset, which we call *initial dataset*, is then used to train our vision and world models. We also construct a control (oracle) dataset, in which the robotic arm either interacts randomly with the environment or pushes the objects in front of it using heuristics. We demonstrate a few instances in Fig. 2.

A. Implementation details

The initial dataset consists of 400 episodes, each containing up to five randomly selected objects. The robot performs 10 push actions per episode and we record $r = 5$ frames of size $(224, 224)$ for each action. The vision encoder is a ResNet18 that outputs $N = 196$ feature vectors of dimension $D_f = 128$. These features are attended by $K = 10$ slots of dimension $D = 128$, refined over $T = 5$ iterations using the Slot Attention mechanism. Each slot is then decoded following the setup in [28]. For the world model, 3D action vectors are first encoded through an MLP and concatenated to each slot representation. These concatenated vectors are then processed using shared edge and node MLPs in a fully connected graph neural network. We train both the vision and world models independently for 100 epochs using a batch size of 16. In the Q-learning framework, we use a ResNet18 as the Q-network, trained for 1000 steps. The policy is optimized using the Double Q-learning objective with an $\mathcal{L}_1$ loss. The target network is updated via an exponential moving average with $\tau = 0.005$. Key architectural choices and hyperparameters are summarized in Table III.

B. Training the vision and world models

We train the vision model with the Adam optimizer using a multi-step learning rate schedule. As shown in Fig. 1, the model effectively learns to bind distinct slots to different objects and reconstructs the input frames adequately well.

For the world model, we adopt a training procedure similar to that of [32], keeping the vision components (encoder, Slot Attention, and decoder) frozen. However, due to the limited diversity of the dataset, collected under random action policies, the world model struggles to develop a meaningful understanding of object dynamics. In most episodes, the only moving element is the gripper, which biases the model toward learning only *ego-motion*. As a result, it fails to capture or predict the movement of objects when interacted with, as illustrated in the third column of Fig. 4.

C. Collecting informative trajectories

With a functional understanding of the environment in place, we proceed to actively collect informative data. At this stage, the vision model is capable of reasonably localizing objects, and the world model can predict the gripper's motion with fair accuracy. These capabilities are sufficient to drive a policy that seeks novel and informative trajectories, however standard curiosity signals such as the vision model's reconstruction loss, the world model's raw prediction error [18], or even the disagreement among an ensemble of world models [19] tend to produce noisy or unstable rewards in our visually complex setting, making them ineffective for training a reliable exploration policy. Instead, our intrinsic reward signal, designed to filter out noise from the partially trained vision and world models, successfully reflects the world model's predictive uncertainty.

The policy is trained using Double Q-learning [33], for 1000 steps, with a classic decaying epsilon-greedy exploration strategy. The Q-network is a ResNet that takes as input the 10 segmentation masks $\{\mathbf{\Pi}_k\}$ concatenated in the channel dimension. Interestingly, the learned policy predominantly suggests pushing actions that cause *object motion*. We quantify this observation by measuring that the average displacement of object centers of mass is **three times greater** than under random actions. We also demonstrate this qualitatively in Fig. 3 by visualizing the positions of the best actions that the learned policy suggests.

D. Enhancing World Perception and Modeling

We leverage the learned, intrinsically motivated policy to collect a new dataset composed of more diverse and dynamic interactions, particularly involving object motion. As before, the new dataset contains 400 episodes of 10 actions each.

We initialize the vision and world models with parameters from the initial training and fine-tune them for 100 epochs using a reduced learning rate and the same training setup.

As shown in Tables I, II, fine-tuning on this enriched dataset leads to substantial improvements in both reconstruction and predictive accuracy, compared to models trained exclusively on data from random actions. To showcase this, we evaluate the models on the control dataset; namely on newly, independent, collected test trajectories generated using either random or heuristic action policies. In more detail, Table I shows that the re-trained vision encoder provides more balanced results across the test sets, compared to the action-biased alternatives. The vision model can now reconstruct better the instances corresponding to heuristic actions, while maintaining its performance on instances of random actions. In practice, we want good reconstruction to both random and heuristic data to reflect our ability to "perceive" our world.

Moreover, in the fourth column of Fig. 4 we demonstrate that the world model is now able to predict the movement of the object when pushed by the gripper, which is quantitatively shown in Table II. Here, we consider two enhancement pipelines: 1) keep the initial vision encoder frozen and fine-tune the world model and 2) use the previously enhanced vision (EV) encoder frozen and fine-tune the world model. As expected, the enhanced vision encoder further improves the reconstruction metric. Note that both enhancement versions, provide better results compared to the considered biased alternatives, even with respect to the system trained on heuristic actions.

E. Discussion

These results highlight the importance of data quality in object-centric learning and further support the notion that the models themselves can be used to drive the collection of more informative trajectories. This can be seen as an iterative process: *discover informative trajectories of high reconstruction uncertainty* and *minimize the uncertainty by fine-tuning the vision and the world models*. The effectiveness of this iterative process is out of the scope of this paper. However, it paves the way for more intricate world exploration and adaptation; for example one can progressively add new set of objects into the scene, let the robot interact with them and eventually learn their vision attributes, as well as their dynamics.

Our approach relies heavily on the object-centric nature of Slot Attention. In principle, we could train a vision encoder based on, for example, Masked Autoencoders (MAE) [35] on our dataset and use its features both as input to the world model and as state representations for the RL pipeline. However, in practice, this approach proved unstable, likely due to the limited visual diversity in our dataset. Moreover, the object-centric structure we adopt simplifies both the RL and world modeling problems: the policy operates over high-quality segmentation masks and the world model's graph is defined over compact slot representations rather than unstructured visual embeddings.

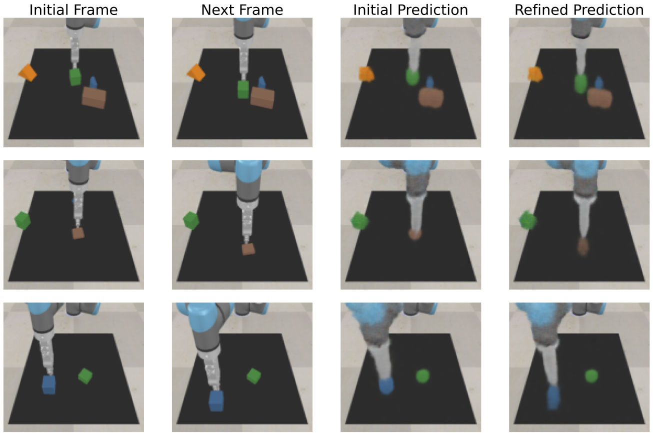

Fig. 4: Qualitative Evaluation: The first and second columns show image frames before and after an action. The third and fourth columns display the predicted motion from the world models trained on the initial dataset and on the new, informative trajectories, respectively. Note the improved accuracy of the refined model in capturing object movement.

V. Conclusions and Future Directions

This work explores whether a robot can adequately perceive its surrounding and their dynamics purely through interaction, without prior knowledge. Drawing from cognitive science, which links perception to predictive models and sensorimotor learning [36], [37], we design a self-supervised, object-centric framework inspired by how infants interpret and interact with the world [38]. Our method improves perception and prediction by fine-tuning vision and world models on data collected via an intrinsically motivated policy. This results in significant gains in both reconstruction and dynamics modeling.

Future work could extend this approach to continual learning, where models are updated online as new experiences are collected, rather than separating data collection and model training into distinct stages. Another promising direction is to explore multiple intrinsic objectives, such as maximizing visual entropy, to encourage more diverse exploration. Additionally, the extracted slot representations could be leveraged for downstream control tasks, enabling us to quantify how well our method supports sample-efficient policy learning. Finally, deploying the framework on a real-world robotic system in a fully autonomous setting, with integrated navigation, interaction and concurrent exploration of spatial and dynamic aspects of the environment, would be a significant next step toward embodied intelligence.

Trained On	Random Test Data	Heuristic Test Data
Random Actions	**0.062**	0.084
Heuristic Actions	0.070	**0.053**
Intrinsic Reward Policy	0.063	0.069

TABLE I: Evaluation of Vision Models on Different Test Datasets We report the reconstruction loss (MSE x10^{-2}) across three configurations. A vision model trained on random actions, one trained on heuristic actions, and one trained on the dataset collected using the intrinsically motivated policy.

Trained On	Random Test Data	Heuristic Test Data
Random Actions	0.209	0.235
Heuristic Actions	0.247	0.164
Intrinsic Reward Policy	**0.131**	0.143
Intrinsic Reward Policy (EV)	**0.131**	**0.135**

TABLE II: Evaluation of World Models on Different Test Datasets We report the reconstruction loss (MSE x10^{-2}) for the predicted next frame across four configurations: a world model trained on random actions, one trained on heuristic actions, and two trained on the dataset collected using the intrinsically motivated policy, either with the initial vision model or the enhanced vision model (EV).

TABLE III: Model architectures and key hyperparameters

Component	Submodule / Setting	Details
Vision Model	Encoder Slot Attention Decoder	ResNet18, $N = 196$, $D_f = 128$ $K = 10$, $D = 128$, 5 iterations As in [28]
World Model	Structure Action MLP Node MLP Edge MLP	GNN with fully connected graph [3, 64, 16] [256, 128, 128, 128] [256+16, 128, 128, 128]
Policy	Network Action Selection Policy Training	ResNet18, output: spatial Q-map ε-greedy ($\varepsilon = 0.9 \rightarrow 0.05$, decay = 1000) L1 Loss, EMA update $\tau = 0.005$
Training	Optimizer Batch Size Epochs	Adam, LR: $1e^{-4}$ 16 (vision/world/Q-learning) 100 (vision/world), 1000 Q-learning steps

VI. ACKNOWLEDGMENT

This project is funded by the European Union under Horizon Europe (grant No. 101136568 - HERON).

Funded by the European Union

REFERENCES

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in *Advances in Neural Information Processing Systems 25*, 2012.

[2] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *Nature*, 2015.

[3] A. Gopnik, “Scientific thinking in young children: Theoretical advances, empirical research, and policy implications,” *Science*.

[4] C. Kidd and B. Y. Hayden, “The psychology and neuroscience of curiosity,” *Neuron*, 2015.

[5] Y. LeCun, “A path towards autonomous machine intelligence,” 2022.

[6] A. Baranes and P.-Y. Oudeyer, “Active learning of inverse models with intrinsically motivated goal exploration in robots,” *Robotics and Autonomous Systems*, 2013.

[7] V. G. Santucci, G. Baldassarre, and M. Mirolli, “Grail: a goal-discovering robotic architecture for intrinsically-motivated learning,” *IEEE Transactions on Cognitive and Developmental Systems*, 2016.

[8] C. P. Burgess, L. Matthey, N. Watters, R. Kabra, I. Higgins, M. M. Botvinick, and A. Lerchner, “Monet: Unsupervised scene decomposition and representation,” *CoRR*, vol. abs/1901.11390, 2019.

[9] K. Greff, R. L. Kaufman, R. Kabra, N. Watters, C. Burgess, D. Zoran, L. Matthey, M. M. Botvinick, and A. Lerchner, “Multi-object representation learning with iterative variational inference,” in *Proceedings of the 36th International Conference on Machine Learning, ICML 2019*.

[10] M. Engelcke, A. R. Kosiorek, O. P. Jones, and I. Posner, “GENESIS: generative scene inference and sampling with object-centric latent representations,” in *8th International Conference on Learning Representations, ICLR 2020*.

[11] A. R. Kosiorek, H. Kim, Y. W. Teh, and I. Posner, “Sequential attend, infer, repeat: Generative modelling of moving objects,” in *Advances in Neural Information Processing Systems, NeurIPS 2018*.

[12] J. Jiang, S. Janghorbani, G. de Melo, and S. Ahn, “SCALOR: generative world models with scalable object representations,” in *8th International Conference on Learning Representations, ICLR 2020*.

[13] Z. Lin, Y. Wu, S. V. Peri, W. Sun, G. Singh, F. Deng, J. Jiang, and S. Ahn, “SPACE: unsupervised object-oriented scene representation via spatial attention and decomposition,” in *8th International Conference on Learning Representations, ICLR 2020*.

[14] A. Zadaianchuk, M. Seitzer, and G. Martius, “Self-supervised visual reinforcement learning with object-centric representations,” in *International Conference on Learning Representations*, 2021.

[15] B. Schölkopf, F. Locatello, S. Bauer, N. R. Ke, N. Kalchbrenner, A. Goyal, and Y. Bengio, “Toward causal representation learning,” *Proceedings of the IEEE*, vol. 109, no. 5, pp. 612–634, 2021.

[16] S. Levine, P. Pastor, A. Krizhevsky, and D. Quillen, “Learning hand-eye coordination for robotic grasping with large-scale data collection,” in *International Symposium on Experimental Robotics*, 2016.

[17] D. Kalashnikov, A. Irpan, P. Pastor, J. Ibarz, A. Herzog, E. Jang, D. Quillen, E. Holly, M. Kalakrishnan, V. Vanhoucke, and S. Levine, “Qt-opt: Scalable deep reinforcement learning for vision-based robotic manipulation,” *CoRR*, vol. abs/1806.10293, 2018.

[18] D. Pathak, P. Agrawal, A. A. Efros, and T. Darrell, “Curiosity-driven exploration by self-supervised prediction,” in ICML’17.

[19] D. Pathak, D. Gandhi, and A. Gupta, “Self-supervised exploration via disagreement,” in *Proceedings of the 36th International Conference on Machine Learning*, 2019.

[20] H. Liu and P. Abbeel, “Behavior from the void: Unsupervised active pre-training,” in *Neural Information Processing Systems*, 2021.

[21] J. Kenney, T. Buckley, and O. Brock, “Interactive segmentation for manipulation in unstructured environments,” in *2009 IEEE International Conference on Robotics and Automation*.

[22] P. Fitzpatrick, “First contact: an active vision approach to segmentation,” in *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003)*.

[23] H. van Hoof, O. Kroemer, and J. Peters, “Probabilistic segmentation and targeted exploration of objects in cluttered environments,” *IEEE Transactions on Robotics*, 2014.

[24] L. Pinto, D. Gandhi, Y. Han, Y.-L. Park, and A. Gupta, “The curious robot: Learning visual representations via physical interactions,” in *Computer Vision – ECCV 2016*.

[25] D. Pathak, Y. Shentu, D. Chen, P. Agrawal, T. Darrell, S. Levine, and J. Malik, “Learning instance segmentation by interaction,” in *2018 IEEE CVPR Workshops*.

[26] C. Sancaktar, S. Blaes, and G. Martius, “Curious exploration via structured world models yields zero-shot object manipulation,” in *Neural Information Processing Systems, 2022*.

[27] N. Watters, L. Matthey, M. Bosnjak, C. P. Burgess, and A. Lerchner, “COBRA: data-efficient model-based RL through unsupervised object discovery and curiosity-driven exploration,” *CoRR*, vol. abs/1905.09275, 2019.

[28] F. Locatello, D. Weissenborn, T. Unterthiner, A. Mahendran, G. Heigold, J. Uszkoreit, A. Dosovitskiy, and T. Kipf, “Object-centric learning with slot attention,” in *NeuIPS*, 2020.

[29] K. Cho, B. van Merrienboer, Çaglar Gülçehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, “Learning phrase representations using rnn encoder–decoder for statistical machine translation,” in *Conference on Empirical Methods in Natural Language Processing*, 2014.

[30] N. Watters, L. Matthey, C. P. Burgess, and A. Lerchner, “Spatial broadcast decoder: A simple architecture for learning disentangled representations in vaes,” *CoRR*, vol. abs/1901.07017, 2019.

[31] T. Kipf, G. F. Elsayed, A. Mahendran, A. Stone, S. Sabour, G. Heigold, R. Jonschkowski, A. Dosovitskiy, and K. Greff, “Conditional object-centric learning from video,” in *The Tenth International Conference on Learning Representations, ICLR 2022,*.

[32] T. Kipf, E. van der Pol, and M. Welling, “Contrastive learning of structured world models,” in *International Conference on Learning Representations*, 2020.

[33] H. v. Hasselt, A. Guez, and D. Silver, “Deep reinforcement learning with double q-learning,” in *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

[34] E. Rohmer, S. P. N. Singh, and M. Freese, “Coppeliasim (formerly v-rep): a versatile and scalable robot simulation framework,” in *The International Conference on Intelligent Robots and Systems*, 2013.

[35] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, “Masked autoencoders are scalable vision learners,” in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[36] N. Nortmann, S. Rekauzke, S. Onat, P. König, and D. Jancke, “Primary visual cortex represents the difference between past and present,” *Cerebral Cortex (New York, NY)*, 2013.

[37] L. B. Smith and M. Gasser, “The development of embodied cognition: Six lessons from babies,” *Artificial Life*, 2005.

[38] E. S. Spelke, “Principles of object perception,” *Cognitive Science*, 1990.