Sensory Commutativity of Action Sequences for Embodied Agents: Theory and Practice

Hugo Caselles-Dupré[1,3], Michael Garcia-Ortiz[2], David Filliat[1]

[1]Flowers Laboratory (ENSTA Paris & INRIA), [2]CitAI, SMCSE, City University of London,
[3]AI Lab (Softbank Robotics Europe)
caselles@ensta.fr, mgarciaortiz@softbankrobotics.com, david.filliat@ensta.fr

***Abstract*—Perception of artificial agents is one the grand challenges of AI research. Deep Learning and data-driven approaches are successful on constrained problems where perception can be learned using supervision, but do not scale to open-worlds. In such case, for autonomous embodied agents with first-person sensors, perception can be learned end-to-end to solve particular tasks. However, literature shows that perception is not a purely passive compression mechanism, and that actions play an important role in the formulation of abstract representations. We propose to study perception for these embodied agents, under the mathematical formalism of group theory in order to make the link between perception and action. In particular, we consider the commutative properties of continuous action sequences with respect to sensory information perceived by such an embodied agent. We introduce the Sensory Commutativity Probability (SCP) criterion which measures how much an agent's degree of freedom affects the environment in embodied scenarios. We show how to compute this criterion in different environments, including realistic robotic setups. We empirically illustrate how SCP and the commutative properties of action sequences can be used to learn about objects in the environment and improve sample-efficiency in Reinforcement Learning.**

I. Introduction

Perception is the medium by which agents organize and interpret sensory stimuli, in order to reason and act in an environment using their available actions [1]. We focus on scenarios where embodied agents are situated in *realistic* environments, i.e. the agents face partial observability, coherent physics, first-person view with high-dimensional state space, and low-level continuous motor (i.e. action) space with multiple degrees of freedom.

In classical robotics, we can use a controlled robotic setup where we utilize external information about the agent and the environment, such as position, joint parameters, object positions, and annotated data. This allows the experimenter to distill its knowledge in the form of priors into the system (e.g. knowledge of the workspace in the case of an a robot interacting with objects on a table). However, this information might not be available in the general case. In Nature, children and animals do not have access to this information when they are born. They start from a relatively naive setup, and then build perception via interaction with the environment. We aim at developing theories and applications for this tabula-rasa case where the agent is naive: it can only actuate its motors (without any description of what they do) and receive observations through its sensors.

Embodied agents, when acting in their environment, produce a stream of sensorimotor data, composed of successions of motor states and sensory information. While most current approaches for building perception consider that the interpretation of sensory information is an isolated problem that only requires extracting relevant information in instantaneous sensor values [2], [3], several approaches [4], [5], [6], [7] that can be traced back to 1895 [8], advocate the necessity of studying the relationship between sensors and motors for the emergence of perception.

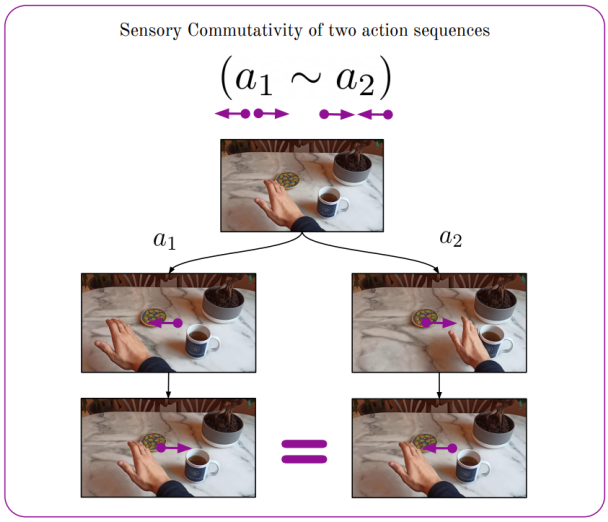

Figure 1: Two action sequences sensory commute if they produce the same sensory state when played in different orders from the same starting position. In this example, the action sequences would not commute if an object would be in the way of the hand movement.

Inspired by these works, we study the commutativity of action sequences with respect to sensors, which we term sensory commutativity, illustrated in Fig. 1. Sensory commutativity occurs when two sequences of actions played in different orders lead to the same final sensory state. In order to study the commutation properties of sequences of actions, we introduce Sensory Commutativity experiments (SC-experiments), which consists in having the agent play an action sequence in two different orders from the same starting point. Sensory Commutativity Probability (SCP) of a degree of freedom is

then a measure of how likely two sequences of actions (in different orders) on this degree of freedom will lead to Sensory commutation. Note that learning SCP is a priori dependent on the environment and the morphology of the agent.

We show that this value has intrinsic meaning for the embodied agent: if the SCP is high then the degree-of-freedom has a low impact on the environment (e.g. moving a shoulder is more likely to lead to environment changes than moving a finger, so SCP for a shoulder is lower than for a finger). By computing the SCP for each degree of freedom of the agent, we are able to characterize its motor space without any a priori knowledge and use this information for subsequent tasks. In our experiments, we illustrate how SCP, and more generally SC-experiments, can be used to learn about objects in the environment and improve sample-efficiency in a Reinforcement Learning (RL) problem. Our contributions are therefore the following:

- We provide a mathematical framework to express sensory commutativity, and theoretical insights on how it can be useful for building perception for an embodied artificial agent.
- We introduce Sensory Commutativity experiments and the Sensory Commutativity Probability criterion: tools based on the commutative properties of action sequences that allow learning about the agent and the environment.
- We provide methods to compute them, including in realistic robotics setups.
- We experimentally show how SC-experiments and SCP can be useful for object discovery and improving sample-efficiency in a RL setup. Our code is available in the supplementary material.

II. Related work and motivation

A. Related work

SensoriMotor theory (SMT) is a theory of perception that gives prominence to the role of motor information in the emergence of perceptive capabilities [9]. Inspired by philosophical ideas formulated more than a century ago by H.Poincaré [8], it led to theoretical results regarding the extraction of the dimension of space [10], the characterization of displacements as compensable sensory variations [11], the grounding of the concept of point of view in the motor space [12], [13], as well as the characterization of the metric structure of space via sensorimotor invariants [14]. The present work studies the commutativity of action sequences with respect to sensory information and takes inspiration from this literature.

An important insight from this literature is that action and sensor spaces have a shared underlying structure, since they are causally linked (sensory changes are caused by actions). It is suggested that the group structure would be well adapted to describe those links [15], [8], yet it has never been formalized in these works. However recently, Symmetry-Based Disentangled Representation Learning (SBDRL) [16], [4] used group theory to formalize disentanglement in Representation Learning using symmetries, i.e. transformations of the environment that leave some aspects of it unchanged. Groups are composed of these transformations, and group actions are the effect of the transformations on the state of the world and representation. Inspired by this approach, we formalize the group structure suggested in the SMT theory and use it to define the SCP criterion.

More generally, the idea of learning how actions influence sensations, and how this information can be used for exploration has been investigated in many ways. A large body of work has investigated developmental robotics [17], [18], with for instance a concept related to the present work called the slowness principle [19]. The idea is that meaningful sensory dimensions change slowly even in the case of rapid actuator changes, which allows identifying meaningful structures such as objects. With the SCP criterion, we actively apply action sequences in different orders and observe the difference in sensors in order to organize useful degrees of freedom of the agent in terms of how much they impact sensors. This general idea is also present in the psychology and neurosciences literature, and is termed proximo-distal principle [20]: the tendency in infants for more general functions of limbs to develop before more specific or fine motor skills. This principle is also visible with the SCP, which allows to explore sensorimotor relations by prioritizing degrees of freedom which lead to bigger sensory changes: fine motor skills have high SCP and general function of limbs have low SCP.

These principles can be applied to acquire meaningful state representations in order to learn how to act in the environment. Our main motivation is to give insights on how sensory commutativity can allow seeing the problem in a novel way. We investigate two applications problems: object detection and sample-efficiency in Reinforcement Learning (RL). For object detection, we either have well-performing methods based on computer vision algorithms and largely annotated databases [21], or algorithms based on data collected by the agent itself [22], [23], [24]. With sensory commutativity, we fall in the second category, as we aim at using sensory commutativity as the tool for detecting objects that the agent can interact with. About sample effiency in RL, the problem is often dependent on representations that are used as states. Most recent solution aim at improving the decision making component of the problem by building a new learning algorithm (HER [25], SAC [26], PPO [27], and many more) which are comparatively better on standard benchmarks. Here, we do not improve the learning algorithm, but rather try to show that by knowing the agent better (by computing its SCP criterion), we can improve sample-efficiency in RL by modifying its exploration strategy.

B. Motivation

Poincaré [8] suggested that the set of compensable transformations of the environment together with the composition operation forms a group, while [15] further attempted at describing this group. Using action sequences and their commutative property, the authors suggested that spatial transformations and non-spatial transformations can be disentangled.

In this paper we build on those previous works by considering the set of action sequences, termed $Seq(\mathcal{M})$, and their commutative properties. We study the group and subgroup properties of $Seq(\mathcal{M})$, with the aim of organizing the

motor space $\mathcal{M}$ hierarchically. This will be achieved with the definition of the Sensory Commutativity Probability criterion.

III. Commutative properties of action sequences

A. Formalism choice

In the SMT theory, the agent sensory motor experience is described as follows:

$$s_t = \phi(m_t, \epsilon_t) \tag{1}$$

This formalism, while close to the RL formalism, is centered around the agent and its perception. At a time t, the agent is in a particular motor state m_t. This means that its motors are in a particular setup called m_t (e.g. the actuator' torque and angle). The environment is defined by everything that's not the agent. It's thus an entity that is in a state ϵ_t, e.g. a room with 6 walls plus light sources and objects placed in different locations. The agent can perceive the world through its sensorimotor dependencies ϕ: a function that takes as input m_t and ϵ_t and produces sensory inputs from its sensors s_t.

Next, we would like to describe the dynamics of the world. This description is generally not present in SMT theory. Thus Eq.1 is not sufficient to support the description of the dynamics of the world. We propose to model these dynamics with the following equation:

$$m_{t+1}, \epsilon_{t+1} = f(m_t, \epsilon_t, \Delta_{m_t}^{m'_{t+1}}, \Delta_{\epsilon_t}^{\epsilon'_{t+1}}) \tag{2}$$

The agent can operate motor commands $\Delta_{m_t}^{m'_{t+1}}$, which will in turn change it's sensory inputs to s_{t+1} through the function ϕ. The environment can change also and influence the agent, represented by $\Delta_{\epsilon_t}^{\epsilon'_{t+1}}$. Taking the initial states and changes as inputs, the function f yields the new motor command m'_{t+1}, and a new configuration of the environment ϵ'_{t+1}. We don't generally have that $\epsilon_{t+1} = \epsilon'_{t+1}$ or $m_{t+1} = m'_{t+1}$ since the agent can affect the environment configuration through its motor commands or the environment can force movements on the agent.

In summary, by combining Eq1 and Eq.2, we obtain an equation that includes the dynamics of the world in classical SMT formulation:

$$s_{t+1} = \phi(m_{t+1}, \epsilon_{t+1}) = \phi(f(m_t, \epsilon_t, \Delta_{m_t}^{m'_{t+1}}, \Delta_{\epsilon_t}^{\epsilon'_{t+1}}))$$

B. Group structure of the set of action sequences $Seq(\mathcal{M})$

We will now formalize groups and sub-groups of symmetries in the case of an agent moving in its environment. We study the set of motor command (or action) sequences of finite length, referred to as $Seq(\mathcal{M})$, and will attempt at describing its structure.

Philipona [15] first defined a relation between action sequences: $h \sim g$ if and only if h and g affect the sensors in the same way. Using our formalism, we can translate this concept into an equality.

Definition 1. *Let $(h, g) \in Seq(\mathcal{M})$. h is equivalent to g under (m_t, ϵ_t), noted $h \sim_{m_t,\epsilon_t} g$ if and only if they produce the same sensory states when applied from the same starting situation of the agent (m_t) and the environment (ϵ_t):*

$$h \sim_{m_t,\epsilon_t} g \iff \phi(f(m_t, \epsilon_t, h, \Delta_{\epsilon_t}^{\epsilon_{t+1}})) = \phi(f(m_t, \epsilon_t, g, \Delta_{\epsilon_t}^{\epsilon'_{t+1}}))$$

Intuitively, two actions sequences are equivalent for a particular motor state and environment state if applying them lead to the same sensory state. For instance in the case of multiple-joints arm moving freely in an empty space, there are multiple different ways of moving the arm from one motor state to another. This yields action sequences $(h_1, .., h_n)$ which are equivalent in this situation (m_t, ϵ_t), we thus have $h \sim_{m_t,\epsilon_t} g$. However in other situations these actions sequences can become not equivalent, for instance if there are objects on the way as illustrated in Fig. 2.

For convenience and clarity, we will drop the notation for dependence on (m_t, ϵ_t) and thus write $h \sim g$ whenever there are no ambiguities in the context. We now consider the structure of $Seq(\mathcal{M})$ under composition $\circ$ with respect to the equivalence $\sim$.

Proposition 1 (Structure of ($Seq(\mathcal{M})$, $\sim$, $\circ$))**.** *The following properties hold:*
1. $\sim$ is an equivalence, i.e. it is reflexive, transitive and symmetric.
2. ($Seq(\mathcal{M})$, $\circ$) is a group w.r.t $\sim$.
3. $\circ$ is not commutative with respect to $\sim$.

Proof. 1) $=$ is an equivalence, thus $\sim$ is an equivalence as well.
2) All 4 properties of the group definition are satisfied. (i) For two action sequences $(h, g) \in Seq(\mathcal{M})$, the composition of h and g is still an action sequence $h \circ g \in Seq(\mathcal{M})$. (ii) $\circ$ is associative with respect to $=$, i.e. $g \circ (h \circ k) = (g \circ h) \circ k$ thus it follows that $g \circ (h \circ k) \sim (g \circ h) \circ k$. (iii) The identity element is the no-op action. (iv) If we suppose that there are no irreversible phenomenons in the environment, then for a fixed (m_t, ϵ_t), all action sequences can be inverted.
3) $\circ$ is not commutative, as we can always explicitly find two action sequences that do not commute. For instance once there exists a movable object in the environment: if the agent is placed left to the object, then let h be moving right and g be moving left. h and g do not commute (Fig. 2).

□

($Seq(\mathcal{M})$, $\circ$) is thus a group w.r.t $\sim$. This structure is consistent with the intuitions in SBRL and SMT theories. In the following, we build on the observation that composing action sequences is not generally commutative as we can measure to which degree they commute. We show how this property can lead the agent to organize and interpret its motor space.

C. Philipona's conjecture

Philipona [15] already studied how action sequences commute with respect to the sensory information received by the agent. Notably, Philipona defined commutative residues. Suppose that an agent doing $h_1 \circ h_2$ leads to a different outcome

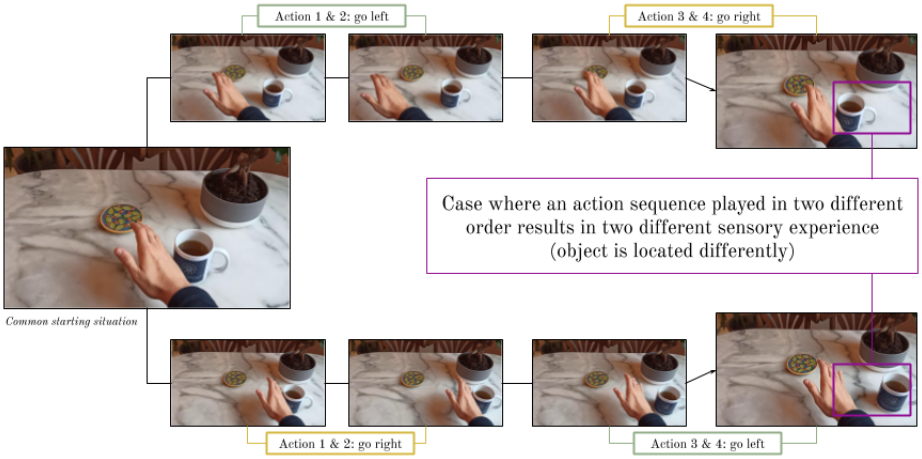

Figure 2: Example of SC-experiment that does not commute. Starting from a common situation, the action sequence played in two different orders does not lead to the same sensory state.

in observations than doing $h_2 \circ h_1$, then a commutative residue g is an action sequence that the agent has to do to compensate the difference in sensory experience.

Definition 2. *g is a commutative residue of (h_1, h_2) if and only if $h_1 \circ h_2 \sim_s h_2 \circ h_1 \circ g$. If g is equivalent to no-op (no action), then h_1 and h_2 commute.*

Starting from this definition, he conjectured that all action sequences that are not displacements commute with any action sequences. For instance, moving your arms (displacement action) then opening the eyes (non-displacement action) will always commute whereas two displacement actions will not necessarily commute, depending on which starting situation s is selected.

Conjecture 1 (Philipona's conjecture)**.** *Let $Seq(\mathcal{M})$ be the set of action sequences. The subset of $Seq(\mathcal{M})$ composed of non-displacements action sequences is the sub-group of $Seq(\mathcal{M})$ that commutes.*

We will illustrate this conjecture with experiments in Sec. IV-C.

D. Sensory commutativity probability of an action sequence

Based on Philipona's conjecture, we derive a criterion for characterizing how much each degree of freedom of the agent affects the world, computable using only sensorimotor data. We define "degree of freedom" (DOF) as a dimension of the multidimensional continuous action space of the agent. We also define what we term a sensory commutativity experiment: for an action sequence h, the agent plays it in two different orders starting from the same situation.

Definition 3 (Sensory commutativity experiment (SC-experiment))**.** *Let h be an action sequence of finite length. Let h_p be a random permutation of h (same sequence but different order).*

We define a sensory commutativity experiment (SC-experiment) as playing h and h_p from the same starting point and comparing the two resulting observations in the agent's sensors.

Using the conjecture, we have that for an SC-experiment, the agent can experience two different sensory outcomes only if the action sequence h is composed of at least one displacement action (an action that affects the environment such as moving limbs or going forward).

However, not all displacement actions are equivalent. The agent is more likely to observe two different outcomes if the action sequence is composed of displacement actions that affect the environment *a lot*. Consider moving your forearm (elbow joint) compared to moving your whole arm (shoulder joint): the latter is more likely to move things around in the environment and thus induce sensory non-commutativity when played in two different orders (i.e. having two different sensory outcomes). An elbow joint should therefore have a higher SCP than a shoulder joint.

We formalize this intuition by defining the Sensory Commutativity Probability (SCP) of a degree of freedom, averaged over all starting situations s:

Definition 4 (Sensory commutativity probability of a degree of freedom)**.** *Let $Seq(\mathcal{M}_k)$ be the set of motor commands (or action) sequences of finite length for the k^{th} degree of freedom of $\mathcal{M}$ (motor state space). Let $h \in Seq(\mathcal{M}_k)$ and let h_p be a random permutation of h (same sequence but different order).*

The Sensory Commutativity Probability of the k^{th} degree of freedom $SCP(\mathcal{M}_k)$ is defined as:

$$SCP(\mathcal{M}_k) = \mathbb{P}_{s,h}[h \sim_s h_p]$$

E. Sensory Commutativity Probability computation

We propose a straightforward procedure to estimate the SCP of each degree of freedom of the agent. We initialize the SCP value to 0 (SCP←0). We then repeat the following process n times for each DOF:

- Sample an action sequence using the selected degree of freedom (a sequence of action where each action is a value between -1 and 1).

- Play it in 2 different orders starting from the same randomly chosen state and save the two final sensor images s_1 and s_2. Compute the distance between the two images $d(s_1, s_2)$.

- Count one (SCP+=1) if $d(s_1, s_2) \leq t$, zero otherwise.

Finally, the estimator of the SCP is the average over the number of trials (SCP←SCP/n).

The parameters of the algorithm are the selected distance function d that allows comparing the agent's observations, the threshold τ, and the number of iterations n. Note that using a simulation allows playing the two action sequences of different orders from the exact same starting position. We discuss the need for simulation to compute SCP and more generally SC-experiments in Sec. VI and how to overcome this requirement for real-life experiments.

F. SC-experiments for object detection

The concept of SCP is based upon comparing outcomes of SC-experiments and evaluating whether the two resulting observations are considered equal or not. Going beyond this equality test, we propose to have a finer analysis of the differences between the two observations obs_1 and obs_2 resulting from an SC-experiment.

Comparing obs_1 and obs_2 leads to three possible outcomes from which the agent can learn about immovable and movable objects in the environment.

- obs_1 and obs_2 are entirely different: the two action sequences from this starting position do not commute, because the agent interacted with immovable objects. Using the position of the agent, we can now map immovable objects in the environment.
- obs_1 and obs_2 are identical: the two action sequences from this starting position commute, because the agent did not interact with anything in the environment (free movement). Using the position of the agent, we know that there are no objects in the current space around it.
- obs_1 and obs_2 are identical except for a limited area corresponding to an object that has been moved: it's the case where the agent has interacted with a movable object that did not block the agent's movement. Hence the two action sequences would have commuted for most of the environment, except for the object that has been moved. We can learn to detect this moving object and track it.

G. Experiments

In order to illustrate all these concepts, the experiments presented in the remainder of this paper are organized as follows: we first show how to compute SCP in 2D simple environments, then in 3D realistic robotic setups. Then, we show how we can use SC-experiments to learn about immovable and movable objects in realistic robotics setups. Finally, we show how SCP can be used for improving sample-efficiency in RL. Our code is attached in the supplementary material.

IV. Sensory Commutativity Probability Experimental Analysis

In this first experimental section, we compute and interpret the SCP in a 2D and a 3D embodied agent scenarios. In order to study the properties of SCP and how it relates to the emergence of the notion of objects, we use simulation environments that have the following properties: embodied agent, navigable space with objects to interact with, first-person high-dimensional observations, low-level high-dimensional action space, and coherent physics.

A. 2D experimental setup

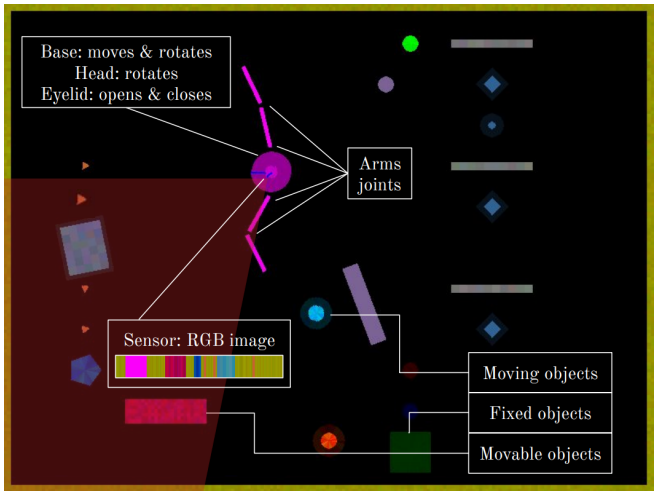

Figure 3: Simulation used for our experiments. The agent Polyphemus has a 8 DOF motor space, receives an image of it's only eye, and is placed in a room with fixed, movable and moving elements.

Simulation description. Our first experiment uses Flatland [28], a platform for creating 2D RL environments. We construct an agent called Polyphemus (a Cyclop from the Greek mythology), that has a base that can move forward and rotate, a rotatable head and two 2-DOF arms. The agent sees through its unique eye that has an activable eyelid, for a total of 8 DOF. The observation received by the agent is a 64x3 line of RGB pixels (as the world is 2D), which corresponds to the field of view of 90 degrees. This agent is placed in a room with fixed, moving, or movable entities, all of different colors. It can move around and physically interact with these entities. Its point of view can change through base movement, rotation, and head rotation. Our simulation is illustrated in Fig. 3. For each degree of freedom, an action or motor command corresponds to a change in the longitudinal/angular velocity of the degree of freedom.

SCP computation. In order to compute the SCP of each of the 8 agent's degrees of freedom (Fig. 4, left), we have

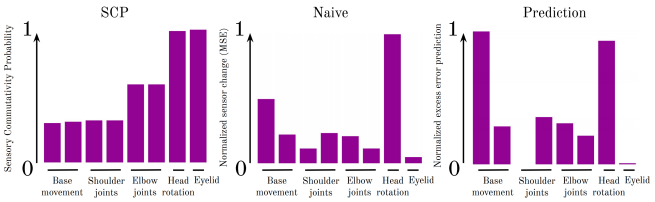

Figure 4: **Left:** Sensory Commutativity Probability for each degree of freedom. **Middle:** Naive alternative. **Right:** Prediction error alternative.

to select a distance and threshold as mentioned in Sec. III-E. The distance selected here is simply the mean squared error between s_1 and s_2, the observations resulting from the two sequences of actions of a SC experiment. Because there is no noise in the dynamics of the environment and the sensor, the future of the agent is deterministic. Therefore, in this particular case we can use a threshold of 0. This means that we consider that two action sequences sensory commutes if and only if applying the two action sequences from the same initial state lead to exactly the same sensors. This hard constrain will be relaxed in subsequent experiments (Sec. IV-B).

Baselines. The SCP criterion derived in this paper estimates how much each degree of freedom affects the environment in an embodied agent scenario. We tried two alternatives to this approach in order to estimate the same quantity. A straightforward approach to this problem, which we call the naive alternative (Fig. 4, middle), is to play action sequences of each degree of freedom and quantify how much the sensors change. A more involved approach is to use prediction on the sensory change caused by each degree of freedom (Fig. 4, right), a common approach used to improve exploration in RL [29], [30]. We call this alternative the prediction error approach. The DOF that are harder to predict could be the ones affecting the environment the most, and thus the most important for manipulation and navigation.

B. 3D realistic experimental setup

We also compute and interpret the SCP for a realistic embodied agent scenario using the interactive Gibson environment (iGibson) [31].

Simulation description. iGibson is a simulation environment for robotics providing fast visual rendering and physics simulation. It is packed with a dataset with hundreds of large 3D environments reconstructed from real homes and offices, and interactive objects that can be pushed and actuated. In our experiments, we use the Rs environment, which is basically a regular apartment. We place the Fetch robot in this environment (Fig. 5, left). Fetch is originally a 10-DOF real robot [32] equipped with a 7-DOF articulated arm, a base with two wheels, and a liftable torso. Fetch perceives the environment through a camera placed in his head (Fig. 5, middle).

SCP computation. In the Flatland environment, two action sequences commuted only if the sensory result of applying both from the same starting situation was perfectly equal. We relax the strict equality condition to compute the SCP for Fetch (Fig. 5, right). Indeed, with real images, only an offset of one pixel would render the two action sequences non-sensory commutative. Instead of using the mean squared error as a distance, we use a perceptual distance using the VGG16 [33] features of each observation. We thus have $d(s_1, s_2) = ||VGG16(s_1) - VGG16(s_2)||_2^2$. The choice of the threshold τ is partly arbitrary, as we are interested in relative comparisons between degrees of freedom. We verify in our experiments that our results and conclusions are valid for a large range of τ.

C. Results

In the Flatland environment, Fig. 4 (Left) shows that only two actions have an SCP of 1: *eyelid* and *head rotation*. All other actions have an SCP inferior to 1. **This is consistent with Philipona's conjecture** (Sec. III-C): *eyelid* and *head rotation* are the two degrees of freedom that are not associated with displacements, thus action sequences composed of actions of these type commute with respect to the sensors. On the contrary, all other degrees of freedom are associated to displacements, and thus will eventually induce non-zero commutation residues when played in different orders from the same starting situation. We observe the same results in iGibson, presented in Fig. 5: the torso lift DOF is not associated with displacement in the environment, so it has an SCP of 1, i.e. it always sensory commutes. Hence the results are consistent with the conjecture and can be used by the agent to autonomously discover which of its actions are associated with displacements or not.

Qualitatively, SCP is inversely proportional to how each degree of freedom affects the environment. By that we mean that from the computation of the SCP, we obtain a hierarchical organization of the action space in which the more important dimensions for manipulation and navigation are separated from the dimensions that are not crucial for such tasks. For instance, we inferred that shoulders should have a lower SCP than elbows since activating the shoulder joint is more likely to induce non-commutativity by moving things around or hitting walls/obstacles. This intuition is verified by our results. Shoulders and base movement have a lower SCP than elbows which in turn have a lower SCP than eyelid and head rotation, as observed in Fig. 4. Without having any prior knowledge about the simulation, we can automatically organize the agent's degrees of freedom in a hierarchy. Moreover, the symmetry of the action space is kept, as elbow 1 and 2 have equal SCP, and so do shoulder 1 and 2. We reach the same conclusions on iGibson (see Fig. 5, right). The wheels have the lowest SCP since they provide longitudinal movement and rotations for the robot. Then comes the first DOF of the articulated arm, i.e. the ones that are closer to its base (like shoulders vs. elbows in the Flatland experiments). Finally, the highest SCP values correspond to the arm DOF that are further on its arm and the torso lift. Once again, we obtain a hierarchical organization of the action space in which the less important dimensions for manipulation and navigation are separated from the dimensions that are not crucial for such tasks.

About the choice of the threshold to compute the SCP, we tried a range of values for τ, from 20 to 100, and in each case,

Figure 5: **Left:** External view of the iGibson simulator where the Fetch robot is in a living room. **Middle:** Fetch's first person view. **Right:** SCP computed for each of Fetch's degrees of freedom.

we obtain the same hierarchy and thus the same conclusion, only the nominal values change, which is irrelevant for the use of SCP.

In additional experiments presented in A, we verified the robustness of these results. We computed the SCP for 8 different combinations of agents and environments (longer/smaller arms, more/fewer objects) and confirmed our intuitions on the interpretation of SCP described above. In additional experiments presented in A, we also verified the robustness of these results in iGibson by computing the SCP for a different type of robot called JackRabbot [34]. We reach the same conclusions as with the Fetch robot.

Alternative methods are not adapted. Details for these two experiments are available in A and results are illustrated in Fig. 4. Both approaches fail to replace the SCP criterion. We see that for the naive approach, rotating the head of the agent changes dramatically what the agent sees, even though this degree of freedom does not affect the environment. For the prediction error alternative, we see the same problem with head rotation and a great difference between the two base movements (rotation and longitudinal movement) while they affect the environment in similar ways. Indeed, it's harder to predict what's outside the field of view of the agent so rotation is harder to predict compared to longitudinal movement. To conclude, the proposed alternatives could not yield the same organization of the agent's DOF.

V. Applications of Sensory Commutativity

A. Sensory Commutativity Probability for object detection

We would like to verify the intuition described in Sec. III-F: there are three possible outcomes to an SC-experiment (different observations, identical observations, and identical observations up to moved objects) and from these outcomes, the robot can detect and map immovable and movable objects in the environment, by doing SC-experiments (playing action sequences in different orders from the same starting point and comparing the resulting observations obs_1 and obs_2). Our experiments are performed in iGibson with the Fetch robot.

1) Method: In order to verify the aforementioned intuition, Fetch needs to be able to perform an SC-experiment and then detect: 1) if the two resulting observations are identical or not, 2) if they are identical except for the parts of an image corresponding to an object that moved. Studies in cognitive science indicate that children are capable of doing this differentiation at a very young age (1 month old) [35], [36], so we consider that equipping the agent with this basic ability is a reasonable assumption. Therefore, we equip the agent with a vision system that gets two observations as input and outputs two masks which will be all zeros if the two observations are identical, all ones if they are different, and the mask of the modified area if something has changed.

We thus train a neural network with generated data to predict those two masks with two observations as input. We refer to this model as the "mask predictor".

Dataset. The data is collected in the Placida environment by starting at a random position in the environment (observation obs_1) and then collecting data for the three possible outcomes:

- no difference: it suffices to keep the same observation and the corresponding masks are all zeros. The data is (obs_1 + all zeros mask, obs_1 + all zeros mask).
- completely different: we move the robot and get a different observation obs_2, the corresponding masks are all ones. The data is (obs_1 + all ones mask, obs_2 + all ones mask).
- no difference except moved objects: we randomly disturb the orientation and position of some movable objects and get a new observation obs_2 identical to obs_1 up the moved objects. The data is (obs_1 + moving objects mask, obs_2 + moving objects mask)

The resulting dataset is illustrated in Fig. 6. The objects we use for training are the original objects found in the interactive Placida environment, augmented with several object from the YCB object benchmark [37].

Architecture and training. We then train the neural network to predict the masks given the observations. This process is similar to predicting the optical flow of two consecutive frames in a video. Thus, for the mask predictor, we compared FlowNet-S [38], a popular baseline for optical flow prediction, with the state-of-the-art RAFT model [39], and selected RAFT because of its higher prediction accuracy. We train the model using the same architecture and optimization process as in the paper, except for the loss function and the output activation function. We change the loss function to a binary cross-entropy loss between the ground truth mask and the output mask of the network. We select the sigmoid function as output activation function so that the model outputs binary masks instead of the original optical flow map output ($2 * W * H$). All training details are available in the original

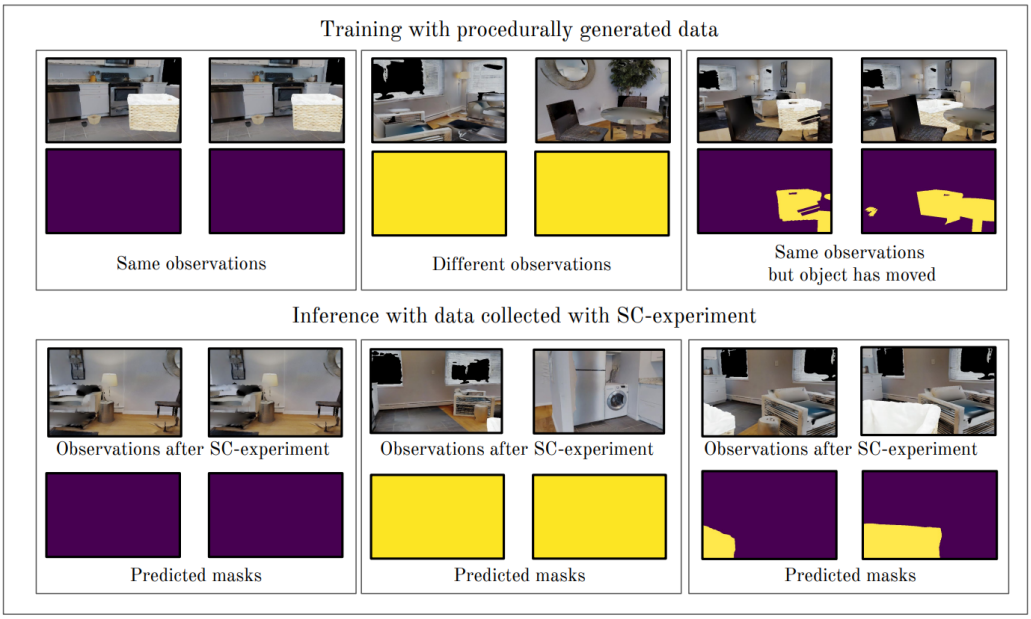

Figure 6: Dataset for training the mask predictor and inference results on data collected with SC-experiments. The dataset is procedurally generated to simulate the three possible scenarii resulting from a SC-experiment. **Left:** scenario where there are no changes in the observations. **Middle:** scenario where the observations are different. **Right:** scenario where the observations are identical up to moved objects.

open-source implementation we used[1].

Inference. Once the mask predictor is trained, we place the agent in an environment and perform SC-experiments where we let it play an action sequence in different orders from the same starting point. Then, the goal is for the agent to detect immovable and movable objects using the generated data from the SC-experiments and the mask predictor. All experimental details are described in A.

Evaluation. For qualitative and quantitative evaluation, we manually create a test set with 50 tuples $(obs_1, obs_2, mask_1, mask_2)$ of the three possible scenario resulting from a SC-experiment. We cannot construct this dataset automatically, as the mask has to be manually created by either assessing if the two observations are different or identifying which object has moved between the two observations. Using this dataset, we can first assess the prediction accuracy among the three possible scenarios.

In the case where an object has moved (see example in lower right corner of Fig. 6), we can further analyze the accuracy of the predicted mask using the Jaccard index, or Intersection over Union (IoU). It quantifies the overlap between predicted $(p1, p2)$ and ground-truth (gt_1, gt_2) masks. It is defined as $IoU_i = \frac{|p_i \cap gt_i|}{|p_i \cup gt_i|}$.

2) Experiments and results: Quantitatively, the performance of the mask predictor on the manually collected test set reach a prediction accuracy among the three possible scenarios of 82%. We reach an average Jaccard index of 0.85 on the subset of instances where an object has moved (see example in lower right corner of Fig. 6).

[1]https://github.com/princeton-vl/RAFT

Do SC-type experiments allow detecting movable objects? We compute SC-experiments using a DOF selected using SCP value. We select the DOF with SCP closest to 0.5 in order for the outcome of SC-experiments to be as diverse as possible, i.e. the DOF of the arm that is closest to the body of the agent.

Results presented in Fig. 6 & 8 show that using the mask detector with the outcome of these SC-experiments allows to detect objects that have been moved. Note that the mask detector only detects objects that have moved between the two resulting observations, rightfully ignoring the other potential objects that were not moved. After this detection, we can then use semi-supervised tracking algorithms such as [40] in order to track the detected object.

Do SC-type experiments allow detecting immovable objects? Results presented in Fig. 6 show that the mask predictor is also able to accurately predict when the observations are different or identical. By isolating those two cases from the case where only one or a few objects have moved, we can compute a local SCP value that tells us whether the agent interacted with an immovable object during the SC-experiment. We can compute this local SCP value for different starting positions in the environment, and then construct a map of immovable objects in the environment. We present this map in Fig. 7 for the arm's DOF that is closer to the body of the agent (we choose this DOF with the same reasoning as the previous result). Results show that regions with low local SCP value correspond to regions where there are walls and immovable objects in the way of Fetch's arm.

Indeed, in the kitchen part (room at the top), the space is cramped and so most of the positions indicate low SCP (less than 0.4) because of the interactions induced with the furniture.

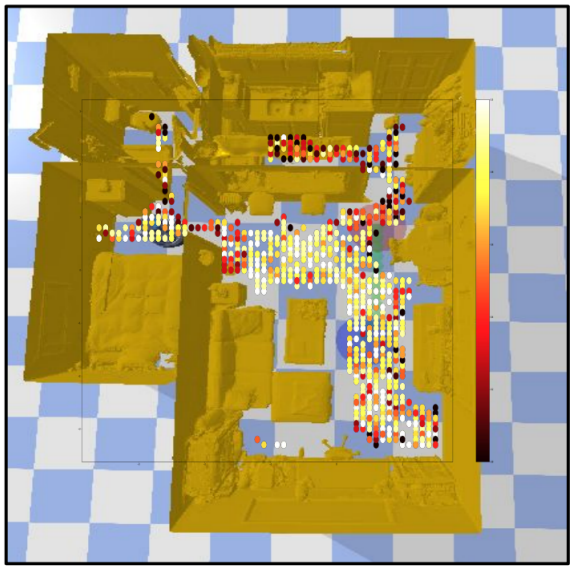

Figure 7: Local SCP value corresponding to the arm's DOF that is closer to the body of the agent, computed over 10 SC-experiments for each position in the Rs-interactive environment.

In the living room (main room) and the bedroom (at the left), most empty space show high local SCP (around 0.8 and 1.0). Notice how the local SCP value is also high around objects that are low and thus have to low chance to interact with the arm (bed in the bedroom, low table in the living room). We thus obtain a mapping of immovable objects in the environment using SCP.

3) Generalization study: In principle, this movable and immovable object detection method is designed to work in any environment, any objects and any field of view. Indeed, it only relies on having a precise mask predictor, which we show can be achieved. We thus performed a generalization study of our method. We performed inference on data with objects, environments and field of view that were not shown during training. For this study, we selected the Rs-environment and the Bolton environment, objects from the YCB benchmark that were not shown during training, and a bigger field of view (90 versus 45 for training).

In Fig. 8, we show results for the generalization study, which indicate that the mask predictor can indeed be used with environments, objects, and field of view that have been not shown during training. Qualitatively, the mask predictor seem to be able to precisely predicts which objects has moved. Quantitatively, the precision of our object detection method in those generalization scenarios is mostly not affected. We manually created another test set of 20 instances with out-of-distribution environments and objects, for the scenario where an object has moved. We reach an average Jaccard index of 0.78, with most instances with a Jaccard index of 1, and a few where the detection is totally missed, thus lowering the average. We observe that when the detection does not totally miss the object, the precision of the mask is excellent. The low performance drop between in-distribution and out-of-distribution test set (0.05) allows us to conclude that our method generalizes to new environments, objects and field of view.

4) Alternatives are not adapted: Alternatives to SC-experiments such as just playing an action sequence and comparing the first and last observations would detect much fewer objects because many experiments would result in a complete image change where the SC-experiments would highlight only a particular object. Another alternative would be to start in a position, play an action sequence, and then go back to this starting point and compare what's changed. While this approach would be comparable for movable object detection, this would not allow detecting immovable objects.

B. Sensory Commutativity for efficient RL

We now illustrate how SCP can be used for unsupervised exploration, by using it to improve sample-efficiency in an RL setup. For computational reasons, we experiment with the Flatland simulator.

1) Experimental setup: We use the PPO2 [27] implementation from Stable-Baselines [41]. The policy is composed of a 1D convolutional feature extractor followed by a recurrent policy. We consider the same agent, Polyphemus, for which we computed the SCP criterion in Fig. 4. The input of the policy is the RGB image of what Polyphemus' eye sees. The environment considered is a square room with 3 dead zones (which terminate the episode with a -20 reward) and a goal zone (which terminates the episode with a +50 reward), illustrated in Fig. 9. We propose two methods that take advantage of the SCP to modify the action space of the agent. The goal is to improve sample-efficiency when learning to solve a task in this embodied scenario.

SCP-truncated action space. We propose to to focus exploration on the degrees of freedom that have a high impact on the environment, by fixating degrees of freedom corresponding to high SCP. We implement this by halving the dimension of the action space, keeping only the degrees of freedom that have the most effect on the environment, i.e. lower SCP value. We thus keep the base movement and rotation, and the shoulders joint, while discarding the elbow joints, head rotation, and eyelid activation. We refer to this method as *SCP-truncated* action space. This action space reduction will simplify the RL task, as long as the necessary actions such as base motion are selected by the SCP criteria.

SCP-adapted action space. A less involved proposition is to modify the action sampling interval according to the SCP value, for each degree of freedom. This method will modify the exploration dynamics to favor important actions. Suppose that the sampling interval for each dimension of the action space is $[-1, 1]$. If a dimension has high SCP, i.e. it does not affect the environment a lot, we then reduce the interval from which actions are sampled $[-1 \cdot l(SCP), 1 \cdot l(SCP)]$. The function l maps the highest SCP to 0 and lowest SCP to 1, then we use a linear interpolation between those two points to deduce values for $SCP \in]-1, 1[$. We refer to this method as *SCP-adapted* action space.

Comparison protocol. We compare those two strategies to a baseline policy trained to solve the task with the complete action space. We average the result of each policy over 30 trials initialized with different random seeds, and we test the

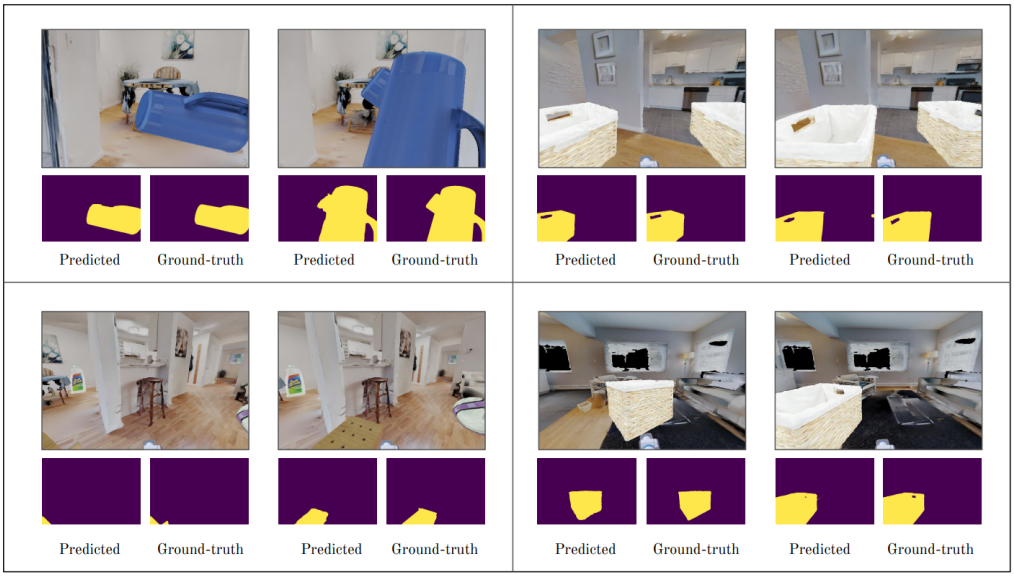

Figure 8: Generalization study of movable object detection using SC-experiments and a mask predictor trained in the Placida environment. In all scenarios, our method correctly predicts the mask object. **Upper left:** Object and environment not seen during training. **Lower left:** Object, environment and field of view not seen during training. **Upper and lower right:** Field of view not seen during training.

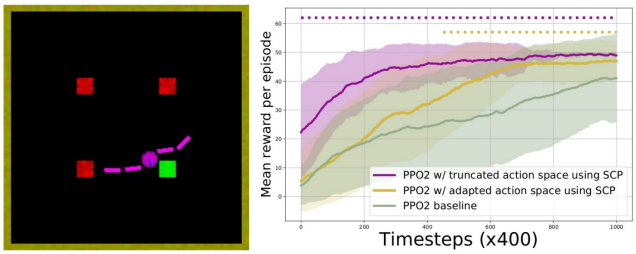

Figure 9: **Left:** RL task. **Right:** Results.

statistical significance of our results according to the guidelines provided by [42].

2) Results: The results are displayed on Fig. 9. First, we notice that all strategies are viable to solve the task. We now compare sample-efficiency between the strategies. The policy trained with *SCP-truncated* action space can learn how to solve the task more than twice as fast as the baseline policy. The discarded degrees of freedom are not crucial in this navigation task, hence the agent is still able to solve the task using only the degrees of freedom that have the lowest SCP value. The policy trained with *SCP-adapted* action space is less sample-effective than the *SCP-truncated* but still learns significantly faster than the baseline policy.

VI. Discussion and Conclusion

Discussion: extending SCP and SC-experiments to real life. The difficulty for SCP and SC-experiments in real-life is that the agent has to be able to play two action sequences from the same starting point. Thus, in a real-life scenario, the method has to overcome stochasticity and irreversible actions (e.g. breaking a glass) which break that assumption. Also, if an object is moved, you would have to place it back to its original position. However, this could be overcome by learning an accurate forward model of the environment that allows the agent to predict what will happen when it plays an action sequence. Consider the forward model as a proxy for one of the experiences. Recent works have made significant progress in this direction [43], [44]. Using this forward model, the agent could play one action sequence and then imagine what would have happened if it had played it in a different order, thus performing an SC-experiment. We believe this is an important future work for using sensory commutativity to build perception for artificial agents, drawing links with the processes of visual attention and surprise [45].

Conclusion. We studied the sensory commutativity of action sequences for embodied agent scenarios. We introduced SC-experiments and the SCP criterion. We showed that SCP is a good proxy for estimating the effect of each action on the environment, for 2D and 3D realistic embodied scenarios. We illustrated the potential usefulness of such criterion and SC-experiments in general by performing movable and immovable object detection and improving sample-efficiency in an RL problem.

References

[1] D. D. Hoffman, "The interface theory of perception," Stevens' Handbook of Experimental Psychology and Cognitive Neuroscience, vol. 2, pp. 1–24, 2018.

[2] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, "Playing atari with deep reinforcement learning," arXiv preprint arXiv:1312.5602, 2013.

[3] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770–778, 2016.

[4] H. Caselles-Dupré, M. Garcia-Ortiz, and D. Filliat, "Symmetry-based disentangled representation learning requires interaction with environments," in NeurIPS, 2019.

[5] A. Laflaquière, "Unsupervised emergence of spatial structure from sensorimotor prediction," arXiv preprint arXiv:1810.01344, 2018.

[6] D. Ghosh, A. Gupta, and S. Levine, "Learning actionable representations with goal-conditioned policies," arXiv preprint arXiv:1811.07819, 2018.

[7] V. Thomas, J. Pondard, E. Bengio, M. Sarfati, P. Beaudoin, M.-J. Meurs, J. Pineau, D. Precup, and Y. Bengio, "Independently controllable features," arXiv preprint arXiv:1708.01289, 2017.

[8] H. Poincaré, "L'espace et la géométrie," 1895.

[9] J. K. O'Regan and A. Noë, "A sensorimotor account of vision and visual consciousness," Behavioral and brain sciences, vol. 24, no. 5, pp. 939–973, 2001.

[10] A. Laflaquiere, S. Argentieri, O. Breysse, S. Genet, and B. Gas, "A non-linear approach to space dimension perception by a naive agent," in 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 3253–3259, IEEE, 2012.

[11] A. V. Terekhov and J. K. O'Regan, "Space as an invention of active agents," Frontiers in Robotics and AI, vol. 3, p. 4, 2016.

[12] A. Laflaquiere, A. V. Terekhov, B. Gas, and J. K. O'Regan, "Learning an internal representation of the end-effector configuration space," in 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 1230–1235, IEEE, 2013.

[13] A. Laflaquière, J. K. O'Regan, S. Argentieri, B. Gas, and A. V. Terekhov, "Learning agent's spatial configuration from sensorimotor invariants," Robotics and Autonomous Systems, vol. 71, pp. 49–59, 2015.

[14] A. Laflaquière, J. K. O'Regan, B. Gas, and A. Terekhov, "Discovering space—grounding spatial topology and metric regularity in a naive agent's sensorimotor experience," Neural Networks, vol. 105, pp. 371–392, 2018.

[15] D. Philipona, "Développement d'un cadre mathématique pour une théorie sensorimotrice de l'expérience sensorielle," 2008.

[16] I. Higgins, D. Amos, D. Pfau, S. Racaniere, L. Matthey, D. Rezende, and A. Lerchner, "Towards a definition of disentangled representations," arXiv preprint arXiv:1812.02230, 2018.

[17] J. Schmidhuber, "Curious model-building control systems," in Proc. international joint conference on neural networks, pp. 1458–1463, 1991.

[18] P.-Y. Oudeyer, F. Kaplan, and V. V. Hafner, "Intrinsic motivation systems for autonomous mental development," IEEE transactions on evolutionary computation, vol. 11, no. 2, pp. 265–286, 2007.

[19] M. D. Luciw, V. R. Kompella, S. Kazerounian, and J. Schmidhuber, "An intrinsic value system for developing multiple invariant representations with incremental slowness learning," Frontiers in neurorobotics, vol. 7, p. 9, 2013.

[20] F. Stulp and P.-Y. Oudeyer, "Proximodistal exploration in motor learning as an emergent property of optimization," Developmental science, vol. 21, no. 4, p. e12638, 2018.

[21] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in Proceedings of the IEEE international conference on computer vision, pp. 2961–2969, 2017.

[22] C. Craye, D. Filliat, and J.-F. Goudou, "Exploration strategies for incremental learning of object-based visual saliency," in 2015 Joint IEEE International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob), pp. 13–18, IEEE, 2015.

[23] N. Lyubova, S. Ivaldi, and D. Filliat, "From passive to interactive object learning and recognition through self-identification on a humanoid robot," Autonomous Robots, p. 23, 2015.

[24] R. Jonschkowski and A. Stone, "Towards object detection from motion," arXiv preprint arXiv:1909.12950, 2019.

[25] M. Andrychowicz, F. Wolski, A. Ray, J. Schneider, R. Fong, P. Welinder, B. McGrew, J. Tobin, O. P. Abbeel, and W. Zaremba, "Hindsight experience replay," in Advances in neural information processing systems, pp. 5048–5058, 2017.

[26] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," arXiv preprint arXiv:1801.01290, 2018.

[27] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," arXiv preprint arXiv:1707.06347, 2017.

[28] H. Caselles-Dupré, L. Annabi, O. Hagen, M. Garcia-Ortiz, and D. Filliat, "Flatland: a lightweight first-person 2-d environment for reinforcement learning," arXiv preprint arXiv:1809.00510, 2018.

[29] Y. Burda, H. Edwards, A. Storkey, and O. Klimov, "Exploration by random network distillation," arXiv preprint arXiv:1810.12894, 2018.

[30] D. Pathak, P. Agrawal, A. A. Efros, and T. Darrell, "Curiosity-driven exploration by self-supervised prediction," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 16–17, 2017.

[31] F. Xia, W. B. Shen, C. Li, P. Kasimbeg, M. E. Tchapmi, A. Toshev, R. Martín-Martín, and S. Savarese, "Interactive gibson benchmark: A benchmark for interactive navigation in cluttered environments," IEEE Robotics and Automation Letters, vol. 5, no. 2, pp. 713–720, 2020.

[32] M. Wise, M. Ferguson, D. King, E. Diehr, and D. Dymesich, "Fetch and freight: Standard platforms for service robot applications," 2016.

[33] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," arXiv preprint arXiv:1409.1556, 2014.

[34] R. Martín-Martín, H. Rezatofighi, A. Shenoi, M. Patel, J. Gwak, N. Dass, A. Federman, P. Goebel, and S. Savarese, "Jrdb: A dataset and benchmark for visual perception for navigation in human environments," arXiv preprint arXiv:1910.11792, 2019.

[35] F. Kaufmann, "Development of motion perception in early infancy," European Journal of Pediatrics, vol. 154, no. 4, pp. S48–S53, 1995.

[36] S. P. Johnson, "How infants learn about the visual world," Cognitive Science, vol. 34, no. 7, pp. 1158–1184, 2010.

[37] B. Calli, A. Walsman, A. Singh, S. Srinivasa, P. Abbeel, and A. M. Dollar, "Benchmarking in manipulation research: The ycb object and model set and benchmarking protocols," arXiv preprint arXiv:1502.03143, 2015.

[38] P. Fischer, A. Dosovitskiy, E. Ilg, P. Häusser, C. Hazırbaş, V. Golkov, P. Van der Smagt, D. Cremers, and T. Brox, "Flownet: Learning optical flow with convolutional networks," arXiv preprint arXiv:1504.06852, 2015.

[39] Z. Teed and J. Deng, "Raft: Recurrent all-pairs field transforms for optical flow," arXiv preprint arXiv:2003.12039, 2020.

[40] S. W. Oh, J.-Y. Lee, N. Xu, and S. J. Kim, "Video object segmentation using space-time memory networks," in Proceedings of the IEEE International Conference on Computer Vision, pp. 9226–9235, 2019.

[41] A. Hill, A. Raffin, M. Ernestus, R. Traore, P. Dhariwal, C. Hesse, O. Klimov, A. Nichol, M. Plappert, A. Radford, et al., "Stable baselines," 2018.

[42] C. Colas, O. Sigaud, and P.-Y. Oudeyer, "How many random seeds? statistical power analysis in deep reinforcement learning experiments," arXiv preprint arXiv:1806.08295, 2018.

[43] D. Ha and J. Schmidhuber, "Recurrent world models facilitate policy evolution," in Advances in Neural Information Processing Systems, pp. 2450–2462, 2018.

[44] D. Hafner, T. Lillicrap, M. Norouzi, and J. Ba, "Mastering atari with discrete world models," 2020.

[45] L. Itti and P. F. Baldi, "Bayesian surprise attracts human attention," in Advances in neural information processing systems, pp. 547–554, 2006.

APPENDIX

The SCP criterion derived in this paper estimates how much each degree of freedom affects the environment in an embodied agent scenario. In this section we discuss why other approaches cannot reliably estimate the same quantity.

Naive approach: changes in sensors. A straightforward approach to this problem would be to play action sequences of each degree of freedom and quantify how much the sensors change. We consider the squared difference for a transition, i.e. the squared difference for two consecutive observations separated by an action sampled from one dimension of the action space. We report the mean squared difference over 100k transitions, for each degree of freedom.

It is clear in our experiment results, shown in Fig. 4, that the approach fails. For instance, rotating the head of the agent changes dramatically what the agent sees, even though this degree of freedom does not affect the environment. It would have made sense if we had considered the top view (fully-observable scenario), since rotating the head does not changes the top view a lot. However in the embodied scenario, this strategy is not viable. For the same reason, approaches based only on the changes in the embodied sensors are bound to fail.

Prediction error approach. A more involved approach would be to use prediction on the sensory change caused by each degree of freedom, a common approach used to improve exploration in RL [29], [30]. The DOF that are harder to predict could be the ones affecting the environment the most, and thus being the most important for manipulation and navigation. We tested this alternative in our experiments, by using a feed-forward neural network to predict the next sensor. The neural network takes a concatenation of the sensor and action at time t and predicts the sensor at time $t+1$. We use the same dataset of transitions as in our experiments with the naive baseline (100k transitions for each degree of freedom, 80k for training and 20k for testing). We trained one model for each degree of freedom, using a neural network with two linear hidden layers with the same number of neurons as the input size. We report the excess prediction error on the held-out test set, i.e. the value of the prediction error minus the minimum prediction error among all 8 degrees of freedom. If the method works, higher excess error prediction should indicate a degree of freedom with more effect on the environment.

The results are shown in Fig. 4. It turns out that prediction error is not well correlated with how much a degree of freedom is important for navigation and manipulation. For instance, head rotation, which does not affect the environment, is hard to predict: the agent might not know what's outside his field of view. On the contrary, base longitudinal movement affect the environment a lot and is easier to predict than head rotation.

To conclude, in our experiments we did not find any viable strategy to replace the SCP criterion. SCP is able to easily estimate how important a degree of freedom is for acting and navigating in the environment. The other considered baselines do not manage to organize the action space in the same hierarchical way.

In our additional experiments on Flatland, we verify some of the intuitions we built with the main experiments on Flatland. For that, we compute the SCP as described in Sec. IV for different combinations of agents and environments. The agents and environments tested are displayed on Fig. 10: we use environments with different numbers of objects (from empty to 12 objects), and two agents: one with longer arms than the other.

The results are also displayed on Fig. 10. Our intuitions are validated since the more objects are place in the environment, the smaller the value of SCP for DOF that correspond to interacting with these objects. For instance in the empty space almost all DOF have a SCP of 1 since there is nothing to interact with but the walls (that's SCP is not perfectly 1 for base movement annd rotation, shoulder and elbow joints).

Also, we notice that if the arms are longer, the SCP for shoulder and elbow joints is consistently lower for each environment. Indeed, there is more chance to interact with objects if the arms are longer, thus inducing a lower SCP.

We follow the same protocol as with the Fetch robot, i.e. we use the Rs environment and the same algorithm to compute the SCP for the 7 degrees of freedom of the JackRabbot: two wheels and a 5-DOF articulated arm. The results are presented in Fig. 11. We observe the hierarchical organization of the DOF of the agent, the wheels having a low SCP as they allow the robot to move around, and the DOF of the articulated arm having a higher and higher SCP as we move closer to the end of the arm (and thus closer to fine motor skills).

We provide further details on the object detection experiments:

- The dataset is composed of roughly 10k instances for each possible outcome (identical, completely different, identical up to moved objects).
- In order to generate the data for the "completely different" outcome, we apply a 90 degrees rotation to the robot.
- For the inference results on movable objects, we experimented with two strategies for the action sequence. Either 20 steps random action sequences or pre-determined action sequences (10 steps where the arm moves left, then 10 steps where the arm moves right).
- For the immovable object detection and creation of the map, we use random action sequences of 100 steps.

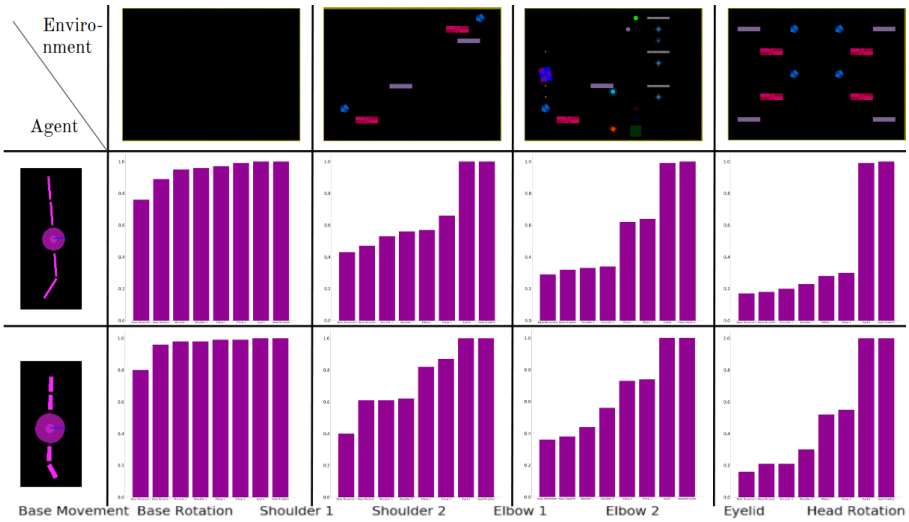

Figure 10: SCP computed for different combinations of agents and environments. Columns: environments. Rows: agents.

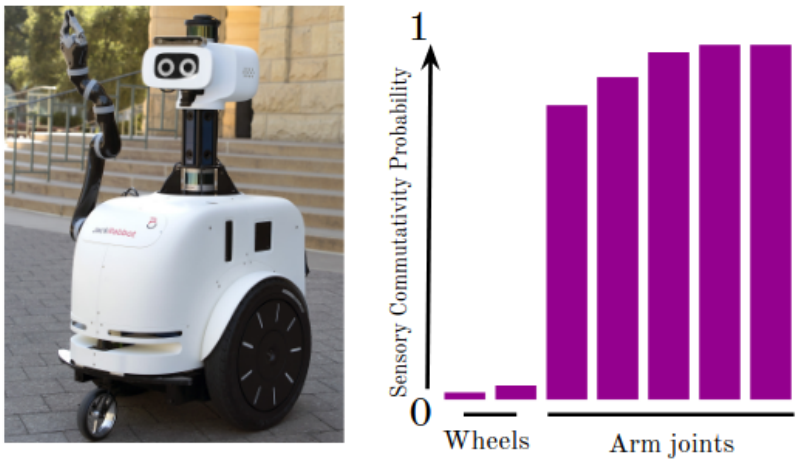

Figure 11: SCP for the JackRabbot (left) in the Rs environment.