ORIGINAL RESEARCH
published: 25 June 2018
doi: 10.3389/frobt.2018.00070

Identification of Invariant Sensorimotor Structures as a Prerequisite for the Discovery of Objects

Nicolas Le Hir[1,2]*, *Olivier Sigaud*[2,3] *and Alban Laflaquière*[1]

[1] AI Lab, SoftBank Robotics Europe, Paris, France, [2] Sorbonne Université, Institut des Systmes Intelligents et de Robotique, Centre National de la Recherche Scientifique UMR 7222, Paris, France, [3] Equipe FLOWERS, INRIA Bordeaux Sud-Ouest et ENSTA Paristech, Talence, France

Perceiving the surrounding environment in terms of objects is useful for any general purpose intelligent agent. In this paper, we investigate a fundamental mechanism making object perception possible, namely the identification of spatio-temporally invariant structures in the sensorimotor experience of an agent. We take inspiration from the Sensorimotor Contingencies Theory to define a computational model of this mechanism through a sensorimotor, unsupervised and predictive approach. Our model is based on processing the unsupervised interaction of an artificial agent with its environment. We show how spatio-temporally invariant structures in the environment induce regularities in the sensorimotor experience of an agent, and how this agent, while building a predictive model of its sensorimotor experience, can capture them as densely connected subgraphs in a graph of sensory states connected by motor commands. Our approach is focused on elementary mechanisms, and is illustrated with a set of simple experiments in which an agent interacts with an environment. We show how the agent can build an internal model of moving but spatio-temporally invariant structures by performing a Spectral Clustering of the graph modeling its overall sensorimotor experiences. We systematically examine properties of the model, shedding light more globally on the specificities of the paradigm with respect to methods based on the supervised processing of collections of static images.

Keywords: object perception, sensorimotor contingencies theory, unsupervised learning, predictive coding, grounding problem

Edited by:
Georg Martius,
Max-Planck-Institut für Intelligente Systeme, Germany

Reviewed by:
Rico Jonschkowski,
Technische Universität Berlin, Germany
Michael Schmuker,
University of Hertfordshire, United Kingdom
Ricardo Sanz,
Universidad Politécnica de Madrid (UPM), Spain

Correspondence:
Nicolas Le Hir
nicolas.lehir@softbankrobotics.com

Specialty section:
This article was submitted to Computational Intelligence, a section of the journal Frontiers in Robotics and AI

Received: *13 December 2017*
Accepted: *25 May 2018*
Published: *25 June 2018*

Citation:
Le Hir N, Sigaud O and Laflaquière A (2018) Identification of Invariant Sensorimotor Structures as a Prerequisite for the Discovery of Objects. Front. Robot. AI 5:70. doi: 10.3389/frobt.2018.00070

1. INTRODUCTION

Humans flexibly interpret their rich sensorimotor experience of the world in terms of objects in the environment. In that respect, we assume that this ability to discover, identify, and manipulate objects is required for any general purpose intelligent robot. Despite great progress in object detection (Redmon et al., 2015) or classification (He et al., 2016) in the last few years, the computer vision community still lacks a clear formalization of the problem of autonomous object identification by an artificial agent. Understanding the fundamental nature of objects and their perception is a core philosophical question that we do not pretend to fully address in this work. Rather, we focus on a specific property that we assume plays an important role in the above question: the spatio-temporal invariance of objects. More precisely, we propose to investigate a mechanism assumed to be fundamental for autonomous object perception, namely the unsupervised identification of invariant spatio-temporal structures in the sensorimotor flow of an agent.

Perception, and in particular artificial perception, is traditionally considered as a passive process in which the sensory state obtained through sensors is projected onto higher-level representations, which in turn inform higher-level cognitive processes which generate actions. This perspective has however been challenged by multiple philosophers and neuroscientists who claim that perceptive experience emerges from internal predictive modeling of the sensorimotor interaction with the environment (Helmholtz, 1896; Gibson, 1979; Friston et al., 2006; Clark, 2013). Our work fits in with such a predictive and sensorimotor description of perception. It is based on two prominent theories, namely the Sensorimotor Contingencies Theory (SMCT) (O'Regan and Noe, 2001) and Predictive Coding (Rao and Ballard, 1999, 2005). The former claims that perception is based not only on sensory information but also on the knowledge of regularities in the way an agent's actions can transform its sensory inputs. The latter suggests that the brain hierarchically builds a predictive model of the causes of its sensory experience. The two viewpoints align nicely when considering that regularities in the sensorimotor flow can be used as support for a predictive model (Seth, 2014).

In this framework, we focus on an elementary property of objects and we study how this property can be exploited to contribute to their discovery by extracting regularities in the sensorimotor experience of an artificial agent. Namely, we assume that objects have an intrinsic structure which is spatio-temporally invariant, and limited in space. In that respect, we assume on the one hand that the intrinsic properties of objects, such as shape, size, or appearance, are preserved across time and space. On the other hand, being limited in space simply means that the objects are smaller than the world explored by the agent.

This spatio-temporal stability of objects implies structure in the sensorimotor experience an agent has when interacting with them. This way, observing one part of a known object, the agent can predict what would be observed on other parts of this object. For example, seeing one side of a tomato, it can predict what the other side of the tomato would look like, as put forward through the concept of *perceptual presence* in Seth (2014). According to the SMCT, this property is constitutive of the experience of objects (O'Regan and Noë, 2001).

In this paper, we propose a minimalistic simulation in which an agent visually explores in a random way an environment containing spatio-temporally invariant structures. We assume having a spatio-temporally invariant structure is one generic property of objects, but it may not be the only one. Hence we refer to identifying these spatio-temporally invariant structures as identifying *proto-objects* in the rest of the paper. Admittedly, as long as the decisions of our agent are random and its actions only consist of visual exploration, we may consider our work from the perspective of pattern identification in signal processing (see e.g., Jain et al., 2000). However, we present this work from an agent-based perspective for three reasons. First, in our framework, an agent is generically "that which acts": it is sufficient that it produces actions to be considered as an agent. Second, in section 3.3.6, we investigate a case where the agent actively rotates objects in its environment. Third, the case of an agent deciding which future action is optimal according to a goal is an important step in our future work agenda.

In our simulations, the world explored by the agent can change in two ways. First, the proto-objects, while keeping their internal structure, can move randomly in the world, or even be introduced/removed. Second, the rest of the environment can itself change randomly. Importantly, despite these changes, the world is statistically invariant enough so that the agent is able to partially explore it between two successive changes. This setup, illustrated in **Figure 1**, can intuitively be interpreted as having proto-objects that can move in the environment, and can be encountered in different contexts. Our model is minimalistic in the sense that we assume no prior knowledge on the world or on the agent itself, neither on its spatial structure, on the environment structure, nor on the proto-objects. The naive

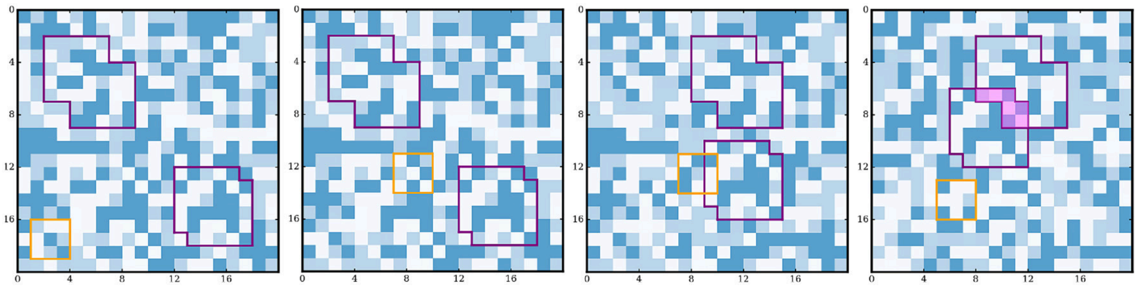

FIGURE 1 | Simulation setup and four example consecutive exploration steps. The position of the agents sensor is outlined in orange, whereas the proto-objects are outlined in purple. The sensor of the agent moves at each time step (but since the null movement is possible, it has a non-zero probability of keeping the same position). At each time step, each proto-object has a probability to move, independently from the other proto-objects and the rest of the environment. At each time step, the rest of the environment has a probability to randomly change. Here for instance, the agent moves at steps 2 and 4. At step 3 both proto-objects move, and at step 4, only one of the them moves, partially overlapping the other proto-object (zone highlighted in purple). At step 4 the environment also changes. Note that the purple outline of the objects is added here for visualization and that the agent does not have any access to it.

agent follows a random exploration policy, and interacts in a generic way with an external environment through an interface of uninterpreted sensorimotor information (Hoffman, 2015; Rafael et al., 2017). In line with Predictive Coding, we propose a method for the agent to build a sensorimotor predictive model of its exploratory experience, and to identify the sensorimotor regularities induced by the proto-objects. More precisely, we model the sensorimotor experience as a weighted multigraph in which the nodes correspond to sensory states, and each pair of states is linked by several edges representing different motor commands. The weight of each edge corresponds to the conditional probability of the corresponding sensorimotor transition. Regularities in the sensorimotor interaction with the environment should then appear as stronger connections between some pairs of nodes. In particular, we hypothesize that the presence of proto-objects should induce the presence of some densely intra-connected subgraphs that the agent can identify as its own experience of these proto-objects. The representation of these regularities can then be used by the agent for counterfactual prediction, which makes the identification of proto-object a worthy objective.

The paper is organized as follows. In section 2, we describe a simple simulation to illustrate the approach, as well as a computational method to identify the sensorimotor regularities induced by proto-objects. In section 3, the results produced by the method applied to the simulated system are thoroughly presented. Additional experiments are also designed to highlight the properties and limitations of the approach. Finally, in section 4, we discuss the benefit of our paradigm with regards to the perception of objects. We also consider the future steps that would extend the current illustrative simulation toward more complex and realistic setups. This work is a direct extension to the preliminary results presented in Laflaquière and Hemion (2015); Hemion (2017).

2. METHODS

In this section, we introduce a simplistic simulation in which an agent explores an environment containing proto-objects. We then propose a method to process its sensorimotor experience and identify the regularities induced by these structures.

2.1. Simulation

The simulation we propose consists in an agent exploring a environment containing proto-objects. The environment is a two-dimensional square gridworld of fixed size 20 × 20 discrete elements, or "pixels." Each pixel can take values in $\{1, 2, 3\}$, and is initialized randomly at the beginning of the simulation. At each time step, the environment can change with a probability $p_{env} = 0.05$, in which case the values of all its pixels are randomly redrawn. At the beginning of the simulation, $N_{obj} = 2$ proto-objects are created in the environment. They correspond to N_{obj} sets of contiguous pixels drawn from the same distribution as pixels of the environment, but which keep the same internal structure during the whole simulation. They are of minimum size 5 × 5 and maximum size 7 × 7, and do not necessarily have a square shape, as illustrated in **Figure 1**. During the simulation, the proto-objects are moved in the environment with a probability $p_{obj} = 0.1$ at each time step. Furthermore, they can be independently removed from the environment with a probability $p_{abs} = 0.2$. If present, the proto-objects pixels occlude those of the environment, and also potentially occlude each other, as illustrated in **Figure 1** . Note that an agent cannot distinguish proto-objects from the environment simply based on a single sensory input, since the pixels that constitute them are drawn from the same distribution. They only differ in the spatio-temporal consistency that proto-objects maintain in contrast with the environment during the simulation.

The agent observes this two-dimensional world with a limited sensor, which is a 3 × 3 patch window, through which it receives sensory inputs. It can move its sensor anywhere in the environment, using motor commands. At each time step, the sensorimotor input of the agent contains a sensory input $\mathbf{s}_t$ (which is a 9-dimensional vector of pixels) and a motor command $\mathbf{m}_t$ (which is a 2-dimensional vector, representing the horizontal et vertical components of the sensor displacement in the visual scene). Together with the sensory input $\mathbf{s}_{t+1}$ experienced after performing $\mathbf{m}_t$, this sensorimotor experience forms a *sensorimotor transition triplet* $(\mathbf{s}_t, \mathbf{m}_t, \mathbf{s}_{t+1})$.

At each time step of the simulation, the agent moves its sensor by randomly picking a new position in the environment (possibly the same as the current one), and stores the experienced sensorimotor triplet. Since the environment and the proto-objects change with a lower probability than the sensor position, the agent can statistically explore their content over several time steps and extract the regularities they induce.

2.2. Processing Method

We now describe the way the agent processes its sensorimotor experience in order to identify proto-objects in the environment. First, the data are compacted by a clustering step. Then, the sensorimotor transitions are stored in a three-dimensional tensor, representing a statistical model of the agent's sensorimotor experience. This tensor is analyzed to extract densely connected subgraphs.

2.2.1. Storing of the Sensorimotor Experience

The agent interacts with the environment during $n_{step} = 3e7$ steps and its sensorimotor experience is stored and processed off-line. We store the empirical conditional probabilities $p(\mathbf{s}_{t+1}|\mathbf{s}_t, \mathbf{m}_t)$ of each sensorimotor triplet $(\mathbf{s}_t, \mathbf{m}_t, \mathbf{s}_{t+1})$ experienced by the agent in a three-dimensional tensor $\mathbf{T}$. In $\mathbf{T}$, $\mathbf{s}_t$ and $\mathbf{s}_{t+1}$ correspond to the row and the column respectively, while m is a one-dimensional encoding of the movement performed at time t and corresponds to the depth in the tensor. However, in order to limit the size of $\mathbf{T}$ and the computational cost of the simulation, the representation of the sensory experience is compacted beforehand by clustering together similar sensations. We use a simple K-MEANS algorithm to perform this clustering, where the number of clusters is arbitrarily set to $N_{km} = 250$. These clusters group together the sensory inputs considered by the agent to build its predictive model, as illustrated in **Figure 2**. In the following, the resulting centroids produced by the K-MEANS clustering algorithm are called "states." The number of possible movements the agent can perform in this 20 × 20 environment is $N_{mv} = 1024$. Thus, the size of the tensor **T** is $(N_{km} \times N_{km} \times N_{mv}) = (250 \times 250 \times 1024)$.

2.2.2. Densely Connected Subgraph Identification

The tensor **T** can be seen as an approximation of the weighted graph mentioned in section 1, in which the weights of the multiple edges between two nodes are the conditional probabilities of the corresponding transition, labeled by the action. We want to identify densely connected subgraphs of sensory states in this graph. To do so, we propose to use Spectral Clustering (Luxburg, 2006; Meila, 2015), which requires the definition of a similarity between each pair of nodes $(\mathbf{s}_a, \mathbf{s}_b)$ in the graph.

Intuitively, two nodes will be considered similar if a transition between them is experienced with a high enough probability. In order to define the similarity, we first filter out some transitions lacking statistical relevance, by discarding rows $T[\mathbf{s}_a, :, \mathbf{m}]$ such that the movement **m** has been performed less than $n_{min} = 20$ times while experiencing state $\mathbf{s}_a$. Then, for other triplets, let $\mathcal{E}$ be the subset of sensorimotor transitions: $\mathcal{E} = \{(\mathbf{s}_a, \mathbf{m}, \mathbf{s}_b) \mid p(\mathbf{s}_b|\mathbf{s}_a, \mathbf{m}) \geq p_{sim}\}$, where p_{sim} is a threshold set to 0.3. For a discussion on the choice of this threshold value, please see section 1.6 of our Supplementary Material. We define the sensorimotor similarity $\Lambda_{\mathbf{sm}}(\mathbf{s}_a, \mathbf{s}_b)$ between each pair of states $(\mathbf{s}_a, \mathbf{s}_b)$ as:

$$\Lambda_{\mathbf{sm}}(\mathbf{s}_a, \mathbf{s}_b) = \sum_{m \in \mathcal{E}} p(\mathbf{s}_b|\mathbf{s}_a, \mathbf{m}).$$

Applying this method to all pairs of states, we derive the 2D sensorimotor similarity matrix $\Lambda_{\mathbf{sm}}$. Finally, since similarities are usually defined for undirected graphs, we make $\Lambda_{\mathbf{sm}}$ symmetric by averaging it with its transpose. This procedure is formally summarized in Algorithm 1. We then apply Spectral Clustering to the graph defined by the similarity $\Lambda_{\mathbf{sm}}$. Spectral Clustering is a graph clustering method that is often used when the relation between the nodes of the graph is quantified by a general measure of similarity, that is not necessary a distance. To define the clusters, the eigenvectors of the Laplacian of the graph are computed. A change of representation is then performed by building new vectors with the components of the main eigenvectors of the Laplacian. A regular clustering is then performed in the space corresponding to these new vectors, yielding the final clusters. More details can be found in Luxburg (2006); Meila (2015).

Algorithm 1 : Building the similarity matrices $\Lambda_{\mathbf{sm}}$ and $\Lambda_{\mathbf{s}}$

```
Data: 3D tensor T of sensorimotor transitions
initialize empty matrices $\Lambda_{\mathbf{sm}}$ and $\Lambda_{\mathbf{s}}$ of size $N_{km} \times N_{km}$ all entries set to 0
for $\mathbf{s}_a$ in $1..N_{km}$ do
    for m in $1..N_{mv}$ do
        $row = \mathbf{T}[\mathbf{s}_a, :, \mathbf{m}]$
        $\Lambda_{\mathbf{s}}[\mathbf{s}_a, :] = \Lambda_{\mathbf{s}}[\mathbf{s}_a, :] + row$
        if $sum(row) > n_{min}$ then
            $p_{max} = max(row)/sum(row)$
            $\mathbf{s}_{b_{max}} = \text{argmax}(row)$
            if $p_{max} > p_{min}$ then
                $\Lambda_{\mathbf{sm}}[\mathbf{s}_a, \mathbf{s}_{b_{max}}] = \Lambda_{\mathbf{sm}}[\mathbf{s}_a, \mathbf{s}_{b_{max}}] + p_{max}$
            end
        end
    end
end
$\Lambda_{\mathbf{sm}} = \frac{1}{2}(\Lambda_{\mathbf{sm}} + {\Lambda_{\mathbf{sm}}}^T)$
$\Lambda_{\mathbf{s}} = \frac{1}{2}(\Lambda_{\mathbf{s}} + {\Lambda_{\mathbf{s}}}^T)$
return Similarity matrices $\Lambda_{\mathbf{sm}}$ and $\Lambda_{\mathbf{s}}$ between sensory states
```

2.2.3. Extracting the Number of Clusters

Spectral Clustering requires the specification of the returned number of clusters. Since we wish to introduce as little supervision as possible in our algorithm, we propose to automatically determine it. There is no universal criterion to automatically determine the relevant number of clusters in a general situation, and most criteria are heuristics (Luxburg, 2006). We propose to use the *cut gap* criterion (Meila, 2015). The cut gap is identified by finding a knee in the curve of the *normalized cut* as a function of the number of clusters. Consider a graph G clustered in N clusters, forming a clustering denoted $\mathcal{C} = (C_1, \ldots, C_N)$. Given $\mathcal{C}$ and a graph similarity Λ_{ij} between each pair of nodes i and j, the normalized cut Ncut(N) is a measure of the quality of $\mathcal{C}$. The lower Ncut, the better the clustering: if Ncut is very low, it means that the clusters are very weakly connected between each other. It is defined as:

$$\text{Ncut}(\mathcal{C}) = \sum_{k=1}^{N} \frac{\text{cut}(C_k, G\backslash C_k)}{d_{C_k}} = \sum_{k=1}^{N} \frac{\sum_{i \in C_k} \sum_{j \in G\backslash C_k} \Lambda_{ij}}{\sum_{i \in C_k} \sum_{j \in G} \Lambda_{ij}}, \quad (1)$$

where the numerator $\text{cut}(C_k, G\backslash C_k)$ is the *cut* between clusters C_k and $G\backslash C_k$, which is a measure of the strength of the connection between C_k and the rest of the graph. The denominator d_{C_k} is the *degree* of C_k, which represents the "weight" of the cluster in the graph. Having low $\text{cut}(C_k, G\backslash C_k)$ terms encourages clusters to be weakly interconnected, while having high d_{C_k} terms favors large clusters, which prevents from yielding trivial isolated outliers as clusters. Thus, the normalized cut leads to a compromise between these two tendencies. In order to find the optimal number of clusters N^*, we automatically detect the largest N which leads to a low Ncut. To do so, we also compute the second order finite difference of Ncut as a function of N,

$$\Delta\,\text{Ncut}(N) = \text{Ncut}(N+2) + \text{Ncut}(N) - 2\,\text{Ncut}(N+1),$$

and we take the value N^* that yields the maximum result, that is: $N^* = \text{argmax}_N\, \Delta\,\text{Ncut}(N)$. Thus, the minimal value that can be returned by this criterion is 2.

2.2.4. Visualizing Predictions From the Tensor

We can also use **T** as a predictive model of the agent's sensorimotor experience. When it receives a certain sensory input, it can use the tensor to try to predict the next sensory input

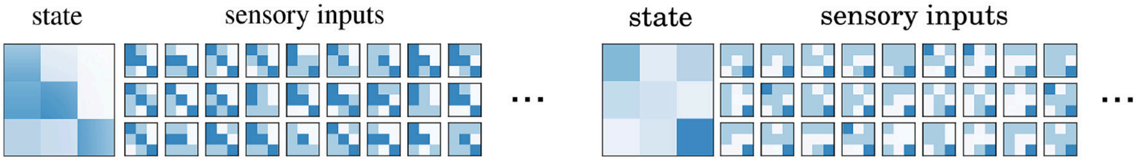

FIGURE 2 | Examples of k-means clustering. Two states and example sensory inputs associated to each of these states by k-means clustering.

for each possible motor command. More formally, say that the agent experiences state $\mathbf{s}_t$ at time t. For each motor command m, the agent has learned a conditional probability distribution on the next state, $p(: |\mathbf{s}_t, \mathbf{m})$, and it can use this distribution to make predictions.

However, the visualization of the predictive model is non-trivial, since the predictions of two distinct motor commands may overlap each other, given that the receptive field of the agent is made of several pixels. In order to illustrate some predictions below, we use a mixture of the distributions, in the following way. Let us consider a pixel position z_{t+1} out of the scope of the agent's sensor. Given that the size of this sensor is 3×3 pixels, 9 motor commands predict the future value of z_{t+1}. We manually average the predictions of these 9 motor commands by computing the weighted average of the states predicted with most certainty by each of these 9 movements. This process requires some knowledge on the sensorimotor structure of the agent, but it is used for illustration purposes only, and not by the agent itself.

3. RESULTS

We now present and analyze the experimental results of the simulations. We then explore alternative setups where the performance of the algorithm is more variable, in order to illustrate the robustness of the approach, but also its limitations.

3.1. Subgraphs Extracted From the Predictive Model

As a reminder, in the simulation, two proto-objects are placed in the environment, with probabilities $p_{obj} = 0.1$, $p_{abs} = 0.2$, and $p_{env} = 0.05$ of movement of the proto-objects, absence of the proto-objects, and of change of the environment, respectively. **Figure 3A** presents the normalized cut Ncut as a function of N, as well as the second order finite difference Δ Ncut(N). The curve of Ncut presents a knee at $N = 3$, and the maximal value of the finite difference is attained for $N^* = 3$. Thus, 3 subgraphs have been identified through Spectral Clustering. This is a good result, since we expect 3 subgraphs to emerge: two subgraphs corresponding to both proto-objects and a third subgraph corresponding to the environment. **Figure 3B** shows the similarity matrix, whose lines and columns have been reordered to group together sensory states belonging to the same cluster. The color of each entry in the matrix corresponds to the similarity between both states. Thus, we can visually see that two subgraphs are strongly connected and a third subgraph is weakly connected.

One can also remark that the probabilities on the diagonal of the matrix are higher than elsewhere. This reflects the stability of the world: there is always a non-zero probability that the agent will encounter two identical sensory states consecutively. Let $p(\mathbf{s}_{t+1} = \mathbf{s}_t|\mathbf{m} = 0)$ be the probability that the agent receives the same sensory input after a time step, given that the agent did not move. If neither of the proto-objects move and the environment did not change, the agent will receive the same input. These conditions being independent but not simultaneously necessary, we can write that $p(\mathbf{s}_{t+1} = \mathbf{s}_t|\mathbf{m} = 0) \geq (1 - p_{obj})^2(1 - p_{env}) \simeq 0.76$. This explains the high values found on the diagonal of the similarity matrix.

3.2. Sensorimotor Prediction

As explained in 2.2.4, the three-dimensional tensor built by the agent can be used as a predictive model of its sensorimotor experience. We illustrate it in **Figure 3C** , for three input states. To clarify visualization, the size of each pixel depends on the probability of each prediction : the largest predicted pixels in the figure are the ones predicted with most certainty. In order to compare the prediction with a ground truth, we also show the ground-truth proto-objects introduced in the environment. If the current state was categorized in one of the densely connected clusters, the model successfully reconstructs the total structure of the corresponding proto-object from the small patch it receives: this is the case for instance for states 24 and 137. On the contrary, for a state categorized in the third, weakly connected cluster, the model predicts no future sensory state with certainty: this happens for instance for state 85.

3.3. Additional Experiments

We propose additional experiments to illustrate the properties and limits of the simulation, the overall approach, and the computational method letting the agent discover proto-objects from its sensorimotor flow.

3.3.1. Importance of the Motor Flow

In order to illustrate the importance of taking the motor commands into account for discovering proto-objects, we propose a similar processing of the experience of the agent, where motor commands are not recorded by the agent. Instead of the sensorimotor similarity $\Lambda_{\mathbf{sm}}$, we derive through Algorithm 1 a *sensory* similarity:

$$\Lambda_{\mathbf{s}}(\mathbf{s}_a, \mathbf{s}_b) = p(\mathbf{s}_a \rightarrow \mathbf{s}_b),$$

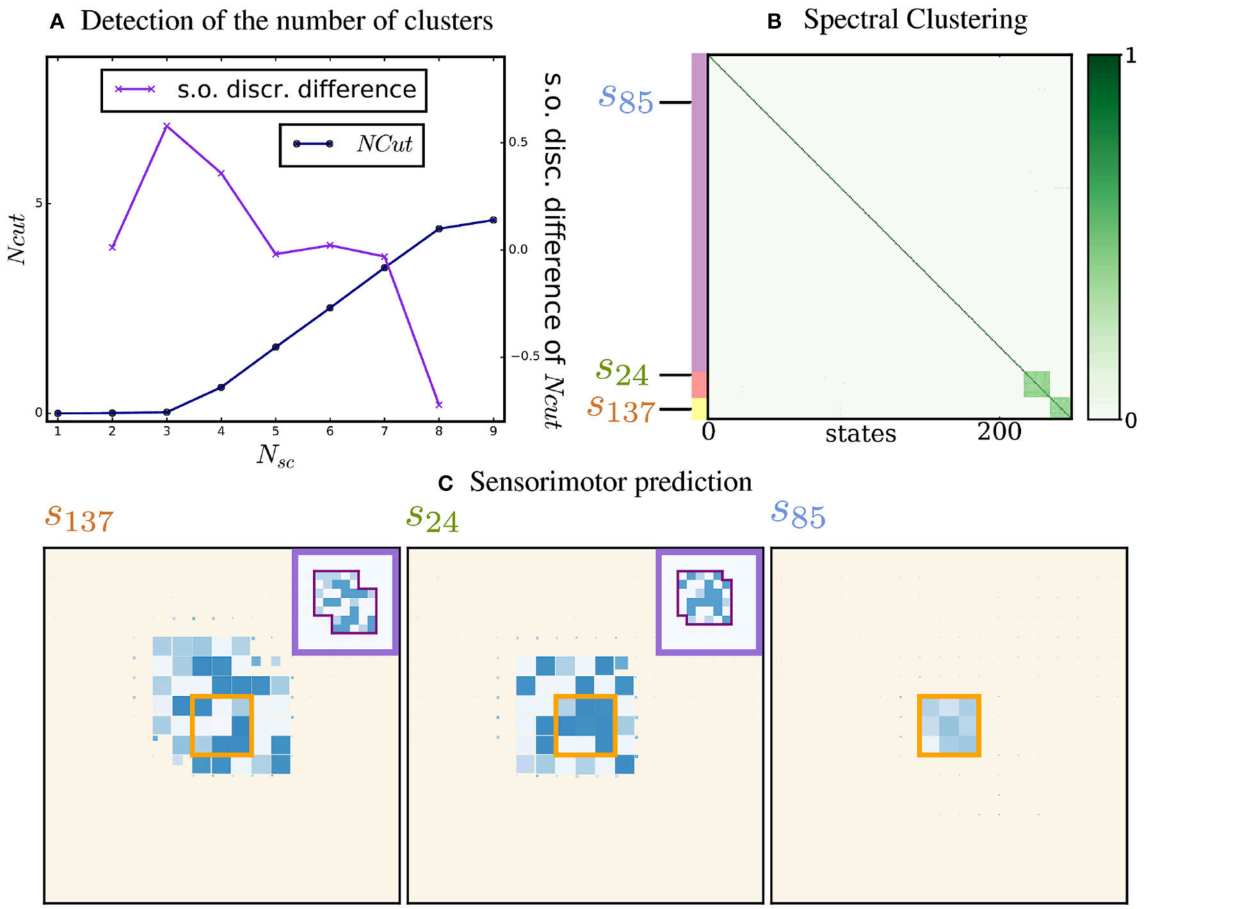

FIGURE 3 | Detection of the number of proto-objects, clustering and prediction. **(A)** Normalized cut and finite difference. N_{cut} (in blue) and second order finite difference of N_{cut}, ΔN_{cut} (in purple) as a function of the number of spectral clusters. A knee in the N_{cut} curve is clearly visible and detected by the second-order derivative at $N^* = 3$. **(B)** Spectral Clustering of the similarity. The rows and columns of the matrix are reorganized according to the clusters. Three clusters are identified: two densely connected ones corresponding to the proto-objects, and one weakly connected corresponding to the environment. The colored strips at the left of the matrix identify the different clusters. The similarity scale is represented at the right of the image. The states that are used to visualize the predictive model are indicated by their number. **(C)** Sensorimotor prediction. Predictive model of the agent for three input states. If the sensory input is classified as part of a proto-object, the agent can predict its future sensory states as a function of its movement (states 24 and 137). If the input state is classified as part of the environment, no probable prediction is made (state 85).

where $p(\mathbf{s}_a \rightarrow \mathbf{s}_b)$ is the probability of transitioning from state $\mathbf{s}_a$ to $\mathbf{s}_b$, regardless of the motor command. Spectral Clustering of this similarity matrix leads to the results shown in **Figure 4**. The agent is no longer able to detect the correct number of clusters. No clear knee in the cut curve is detected and the cut gap criterion returns two clusters, that is the default outcome of our method when it does not find a cut. In **Figure 4B**, we see that this "sensory" similarity matrix, even reorganized, does not display any densely connected subgraph. In our setting, the agent is thus unable to extract the structure of proto-objects without using its motor commands.

3.3.2. Influence of the Number of Proto-Objects

We now propose to study the influence of the meta-parameters of the simulation on the results. We first investigate the impact of the number of proto-objects n_{obj} introduced in the environment on the identification of the densely connected subgraphs. The results are shown in **Figure 5A**. For values up to 4, the number of proto-objects is correctly estimated and the clusters are well defined and densely connected. As the number of proto-objects increases, it becomes harder to detect the correct number of proto-objects. If the number is very large, the sensorimotor experience of the agent contains too much randomness and is poorly predictable, since the proto-objects constantly occlude each other in a random order. As a consequence, the probability of consistently experiencing sensorimotor regularities associated with a given proto-object becomes very low. Here, we see that for $n_{obj} \geq 5$, the Spectral Clustering algorithm does not yield well defined clusters. Note that if the environment was bigger, this overlapping problem would arise for a greater number of

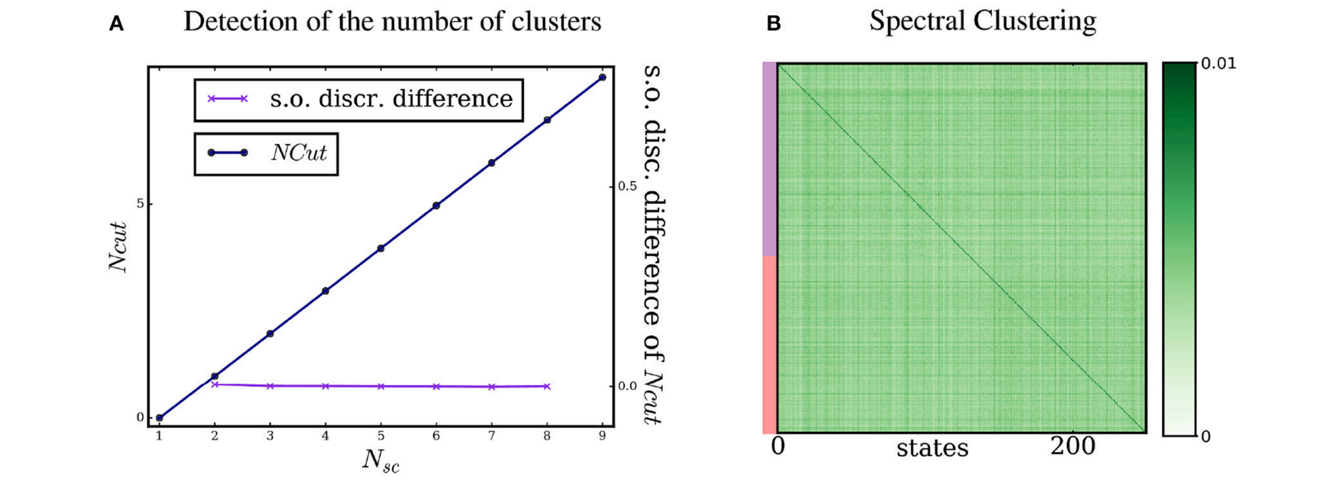

FIGURE 4 | Detection of the number of proto-objects and Spectral Clustering without motor information. **(A)** Applying the normalized cut criterion as presented in **Figure 3** to the sensory similarity $\wedge_{\mathbf{S}}$ yields poorer results. The number of clusters is not correctly detected by the agent. **(B)** Spectral Clustering of the sensory similarity $\wedge_{\mathbf{S}}$ does not present densely nor weakly connected subgraphs. The states seem to be more uniformly weakly connected (note the change in the colormap scale).

proto-objects. It must also be noted that our simulation is not sophisticated enough to properly deal with object occlusions in a consistent way, as a 3D simulation taking the perspective of the agent into account would do. Better dealing with these occlusion issues is left for future work as it requires tackling more difficult questions about memory and the perception of space.

3.3.3. Influence of the Probabilities p_{obj}, p_{env}, and p_{abs}

We investigate the impact of the probability of displacement of proto-objects, p_{obj}, on the result of the clustering. We run the simulation for several values of p_{obj} between 0 and 1, and we show the results in **Figure 5B**. The difficulty of proto-object discovery increases with their probability of movement. This result is expected because the discovery of proto-objects depends on the probabilities of sensorimotor regularities implied by their structure. These regularities vanish when the expected structure cannot be statistically differentiated from randomness, which happens when the proto-objects never keep the same position between time steps. Intuitively, this means that if the world around us were to change constantly, we would not be able to discover objects.

We also investigate the impact of the probability of updating the environment p_{env} on the result of Spectral Clustering. The simulation is run with p_{env} ranging from 0 to 1, with results presented in **Figure 5C**. When p_{env} is high, for instance when $p_{env} = 0.8$, the diagonal of the matrix does not contain high probabilities anymore, since the environment changes too frequently. Although optimal for our simulation, this setup is not realistic considering our own sensorimotor experience, where an environment with no spatio-temporal structure at all is rarely encountered. Another special case arises when $p_{env} = 0$, which means that the environment never changes. Then, the sensorimotor experience while interacting with the environment is completely predictable and the environment should be identified as a third proto-object, as illustrated in the first column of **Figure 5D**. **Figure 6** shows sensorimotor predictions for $p_{env} = 0$. Since the environment never changes, this specific setup highlights sensory ambiguity as one potential limitation of the simulation. Indeed, it is possible for a sensory state to appear in multiple proto-objects, or multiple times in a single object, making it ambiguous. The probability of such a situation is low in the standard setup of the simulation due to the limited size of the proto-objects. However, when $p_{env} = 0$, the whole environment appears as a big proto-object, which significantly increases the probability of encountering ambiguous sensory states. Spectral Clustering is robust to this kind of ambiguity, as it assigns the sensory state to one cluster only, but we can see in the third panel of **Figure 6** that ambiguity can interfere with sensorimotor prediction. Indeed the reference sensory input seems to appear twice in the constant environment.

As a consequence, the sensory prediction of the agent is a mixture of two contributions that overlap. The pixels which correspond to an ambiguous prediction are highlighted in pink. To disambiguate such a situation, the agent would need to have a memory, or a way to hierarchically extract contexts from its sensorimotor experience, as proposed in Hemion (2017).

Finally, we analyze the effect of varying the probability of the proto-objects being absent in the environment. To do so, we run the simulation with changing values of p_{abs} and show the results in **Figure 5D**. Intuitively, the identification of densely connected subgraphs is easier when the proto-objects are present at each time step. On the contrary, it becomes harder when p_{abs} is high, since the sensorimotor regularities associated with proto-objects are encountered with less consistency. Other complementary experiments are presented in the Supplementary Material section of the article.

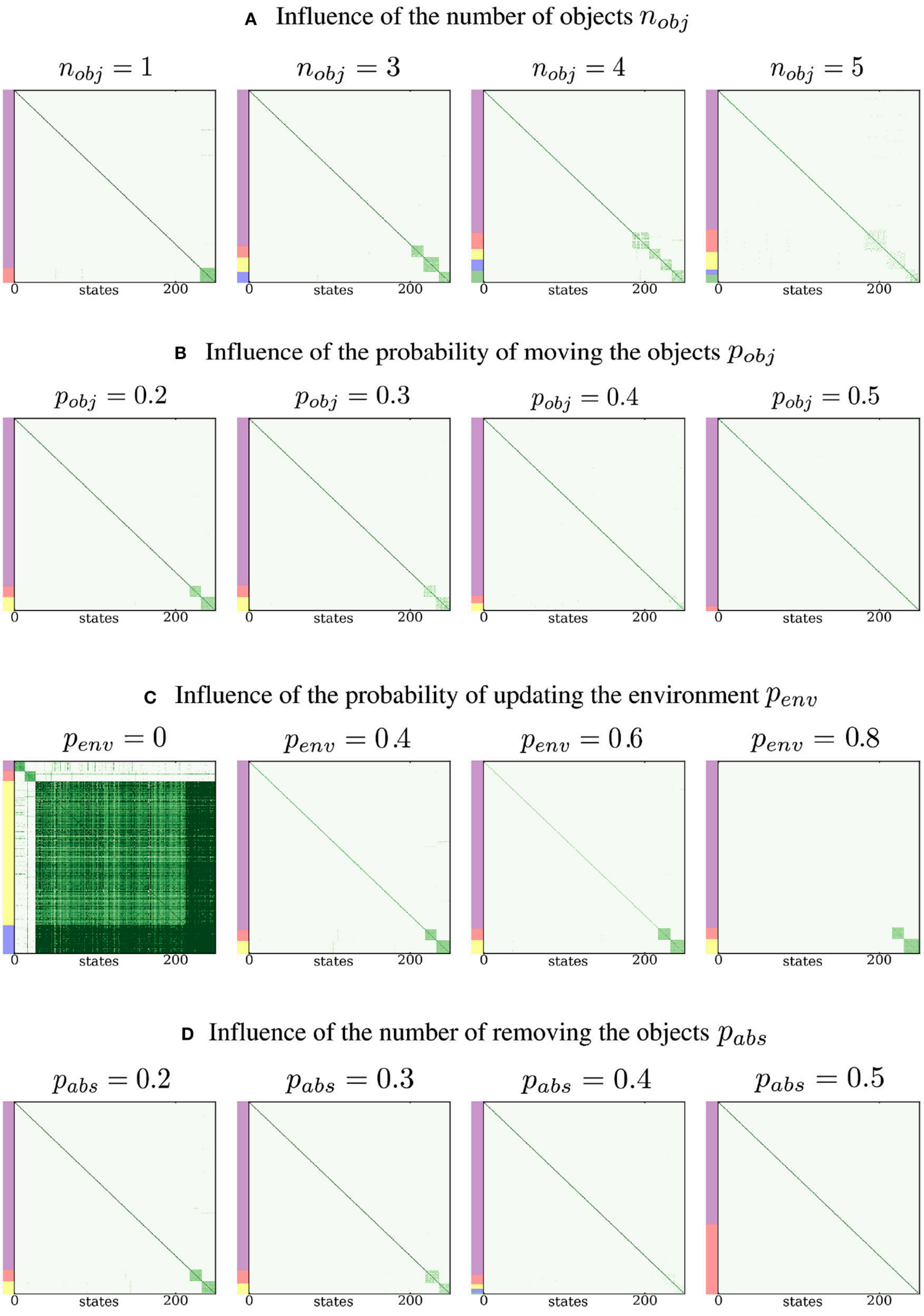

FIGURE 5 | Influence of the parameters of the simulation. **(A)** The agent is not able to discover and differentiate more than 4 proto-objects in this setup. **(B)** When the proto-objects move more frequently, it is harder to extract them from the environment. **(C)** The more the environment changes, the easiest it is to extract proto-objects. When the environment never varies, three proto-objects are identified. **(D)** When the proto-objects are removed too frequently, their extraction fails and the number of clusters is not correctly determined.

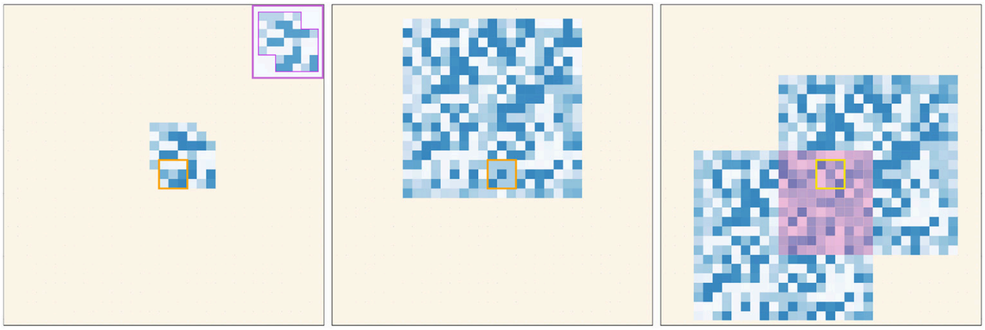

FIGURE 6 | Sensorimotor prediction with a static environment ($p_{env} = 0$). **Left:** prediction of the pixels corresponding to a proto-object. **Center:** the entire environment learned by the agent. **Right:** given the input state, the prediction associated with some specific movements is ambiguous. The corresponding pixels are highlighted in red. For those, several predictions can contradict each other.

3.3.4. Rigidly Linked Proto-Objects

Here we illustrate a property of our definition of proto-objects as spatio-temporally invariant structures. We run a simulation where two proto-objects are rigidly linked: they move together and thus keep their relative spatial position constant during exploration. In **Figure 7**, we see that the agent extracts only one densely connected subgraph. This experience of the agent with two linked proto-objects is thus interpreted as an interaction involving a single proto-object, as the agent extracts a single densely connected graph of sensorimotor transitions. Indeed, the agent looks for sensorimotor regularities without having a notion of spatial contiguity. Thus it does not distinguish the two components of the linked proto-objects. Intuitively this suggests that if we were to live in a world where objects are made of several rigidly linked but disconnected parts, we might interpret them as single entities.

3.3.5. Identical Proto-Objects

We investigate the special case where both proto-objects in the environment are identical instances of the same proto-object. We run the standard simulation where the second proto-object is a copy of the first one, and show the results in **Figure 8**. The agent extracts a single densely connected subgraph. This is expected as the agent cannot separate the sensory inputs coming from one instance of the proto-object from the inputs coming from the other instance. The method can only distinguish types of proto-objects, but not identical instances. A possible solution to separate inputs coming from different instances would be to have a memory and a notion of position in the environment, which the agent does not currently have.

3.3.6. Agent Rotating the Proto-Objects

An important aspect of our approach is that the extraction of proto-objects from the environment should not depend on their visual appearance, which means that it does not depend on their pattern of pixels. Additionally, the actions performed by the agent could be of any nature, meaning they are not limited to sensor movements. In order to illustrate these properties, we run a simulation where the agent can move its sensor and also rotate the proto-objects. This action has no effect on the pixels of the environment, but has the consequence of rotating both proto-objects by 90 degrees. Thus, such a rotation changes the appearance of the proto-objects and the set of sensory inputs that the agent can receive by interacting with the proto-objects is larger than when it cannot rotate them. Results of this simulation are presented in **Figure 9**. After exploration and processing of the sensorimotor data, two densely connected subgraphs are still correctly extracted from the experience of the agent. However, it appears that the clusters are slightly less densely connected than in previous simulations. This might come from the K-MEANS clustering step, since the sensory inputs are distributed differently in the input space, and from the larger number of possible movements.

This shows that if the agent performs non-spatial actions, it can still extract structure induced in its sensorimotor flow by the presence of invariant proto-objects. More generally, any type of action could be performed to learn any structure in the interaction with the world, as long as its effect on the sensory flow of the agent generates some statistical regularities, such as changing the light projected to the global scene, resulting in different pixel values.

3.3.7. Small Proto-Object

We propose a last simulation in which the proto-objects are smaller than the receptive field of the agent. Results are shown in **Figure 10**. No densely connected subgraph is detected by the agent, and it is not able to predict pixels outside the scope of its own receptive field. Since the proto-objects are smaller than the receptive field, the states obtained after the K-MEANS clustering cannot represent the proto-objects accurately, because they also represent pixels that come from the randomly changing environment. Thus, it is likely that these states mix together sensory inputs coming from proto-objects with sensory inputs coming from the environment. Hence, the sensorimotor structure induced by the presence of proto-objects in the world is blurred. Thus, proto-objects smaller than the receptive field cannot be discovered by the agent. A possible way to overcome this limitation could be to consider a set of smaller receptive fields and to process them collectively. This is left for future work.

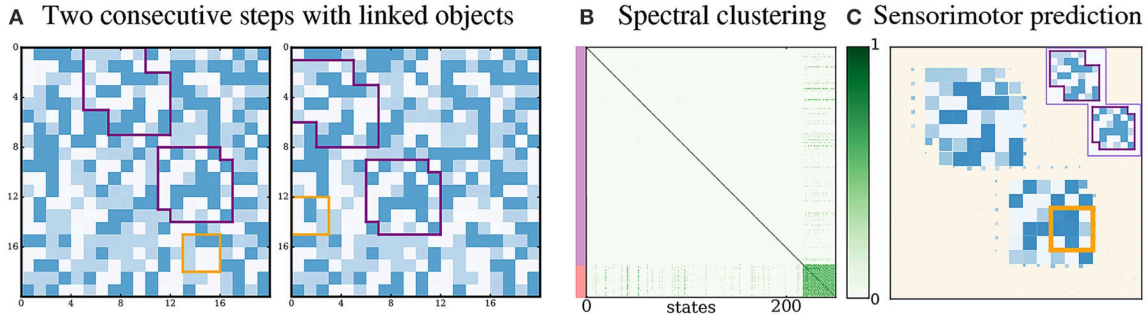

FIGURE 7 | Rigidly linked proto-objects. **(A)** Two consecutive exploration steps when the proto-objects are linked. Between both steps, the proto-objects keep their relative spatial position. **(B)** A single densely connected cluster is identified, corresponding to a single global proto-object. **(C)** Given an instantaneous sensory input from one proto-object, the agent is able to accurately predict states from the other proto-objects if it performs one of the corresponding movements.

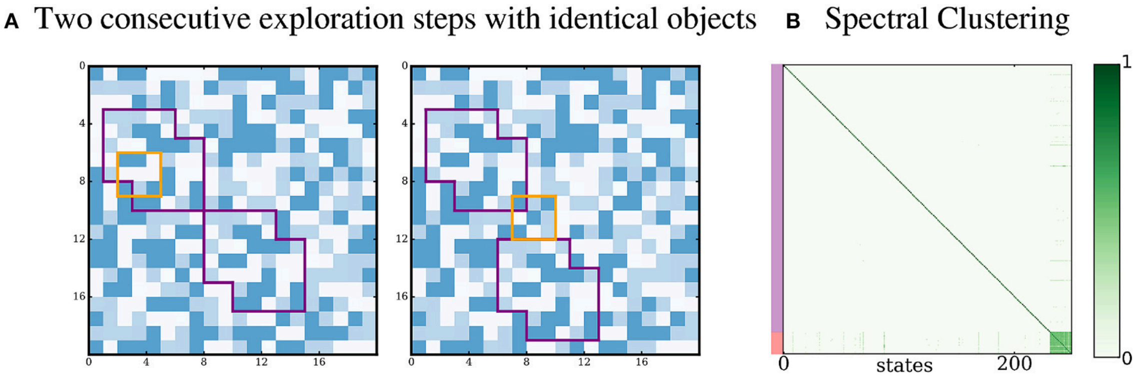

FIGURE 8 | Proto-object with multiple instances. **(A)** Both proto-objects look identical. **(B)** A single densely connected subgraph is identified by the Spectral Clustering step.

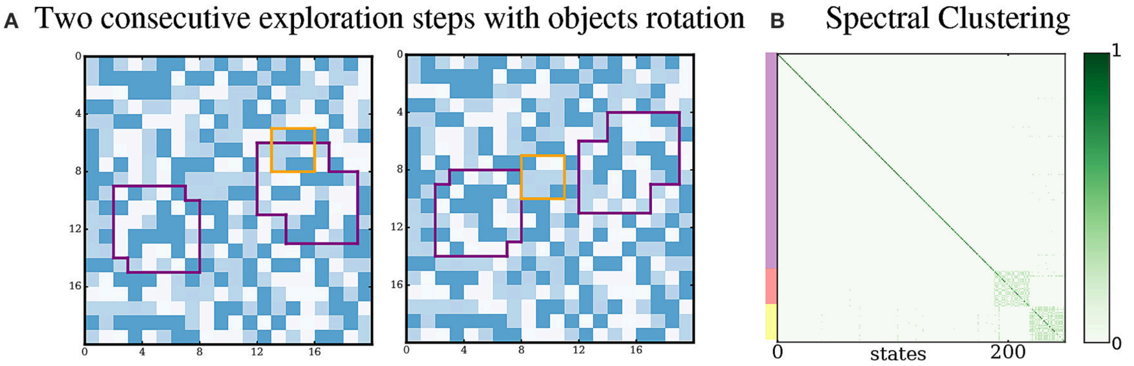

FIGURE 9 | Agent rotating proto-objects. **(A)** Being able to rotate the proto-objects generate additional regularities in the sensorimotor flow of the agent. **(B)** The agent is still capable of discovering them.

4. DISCUSSION

In this work, we addressed object discovery from a sensorimotor perspective. Taking inspiration from SMCT and predictive coding, we defined proto-objects as spatio-temporally invariant structures, that an autonomous agent can detect through regularities in its sensorimotor experience when interacting with its environment. More precisely, the agent discovers such proto-objects by collecting sensorimotor transitions and clustering together sensory states according to a sensorimotor similarity, which we derived from a statistical analysis of those transitions. We illustrated the method by applying it to simplistic simulations and outlined some limitations. We now discuss the specificities of our approach with respect to the standard computer vision

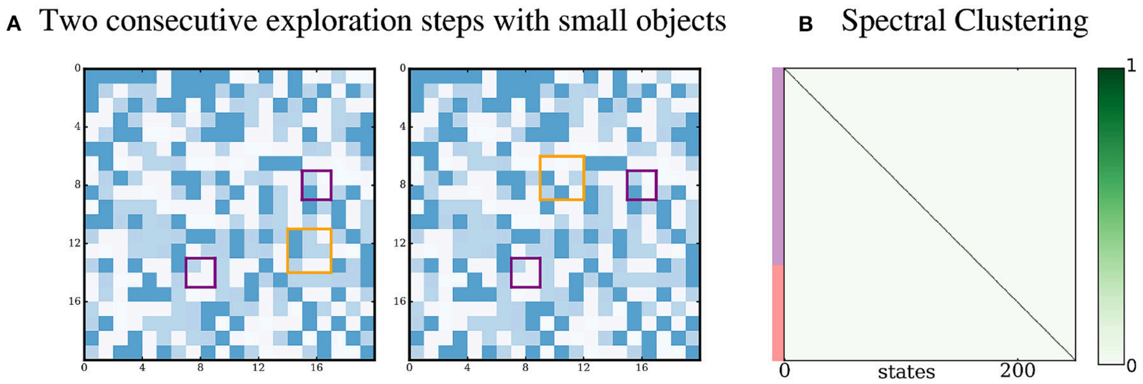

FIGURE 10 | Proto-objects smaller than the receptive field. **(A)** The two proto-objects are smaller than the receptive field. **(B)** No densely connected subgraph is identified by the Spectral Clustering.

paradigm and other related work, we highlight some key properties of the model and we point to future work given the limitations we highlighted.

4.1. Specificities of the Paradigm

In the standard computer vision paradigm, the problem of object identification is generally tackled in a supervised way by training a representation learning algorithm, for instance Deep Convolutional Neural Networks (Gu et al., 2015). These algorithms are trained on a large database of static images containing objects, where the identity of the object is provided as a label (see for instance the well-known ImageNet database Deng et al., 2009). Labeling such databases requires a large human effort which can be mitigated by using semi-supervised or transfer learning approaches, without fundamentally changing the underlying object perception paradigm. In this paradigm, identifying an object consists in extracting from a collection of static images of the same object some invariant set of visual features which are sufficient for discriminating this object from any other. From an engineering point of view, this paradigm is quite efficient as it provides a working solution for many concrete applications. From a more fundamental standpoint, it captures some important aspects of perception in terms of invariant visual features which are not captured in our work. But this paradigm goes with some issues, as revealed for instance by failure on adversarial examples (Szegedy et al., 2013). Another well-known issue is that relying on external labels makes the agent limited to the recognition of objects present in the database. In that respect, using unsupervised learning methods is mandatory if one wishes to design a truly autonomous learning agent. Our work reveals a third issue. Indeed, any approach processing static images individually cannot extract any object from our simulations since the distribution of pixel values is the same in proto-objects and in the environment. Thus in our work, we are not interested in the visual features characterizing the appearance of an object, but rather in its spatio-temporal consistency.

Thus our approach is focused on a property of objects that is orthogonal to the one captured by the standard computer vision paradigm. Instead of focusing on the extraction discriminative spatial features in static images, we focus on extracting spatio-temporally invariant patterns in the sensorimotor flow of the agent. Our approach has several assets. First, it is unsupervised, as opposed to most approaches to the problem of objects detection and classification outlined above. The agent relies neither on externally provided labels nor on rewards, and does not solve a specific task. It discovers the presence of proto-objects, fundamentally driven by the prediction of its sensorimotor experience, and without knowing the structure of these proto-objects in advance. The agent has prior knowledge neither on its sensory structure, nor on the environment, and not even on the structure of the proto-objects: their number, sizes, shapes, appearances, and positions are unknown.

Importantly, the interaction of the agent with the environment does not have to be spatial: the actions performed do not have to be spatial displacements, such as the translations of the sensor as used in the simulations, and the agent does not need to know its spatial position. More generally, the actions performed by the agent and their effects in the environment can be of any nature, as long as they remain consistent in time. As an example, we have shown in **Figure 9** that the agent can extract proto-objects performing actions that modify their appearance by rotating them. The determination of the class of actions that are necessary and sufficient to build an artificial object perception system following our approach is an important and open question, left for future work.

4.2. Related Work

There are other approaches to the problem of artificial perception that exploit either unsupervised learning, the temporal information in the sensory flow of an agent, or the interaction between an agent and its environment.

Unsupervised learning algorithms typically capture statistical structure in the data in order to compress them, hopefully creating more abstract representations (Bengio et al., 2013). Despite some interesting attempts around generative models (Doersch et al., 2015), it is still unclear how such statistical method applied to static images could lead to the development of a complete autonomous perceptual system. Most of the time, although pretraining a neural network in an unsupervised way can be used to bootstrap a supervised learning system (Erhan et al., 2010), the representations built this way are interpreted a posteriori by a human.

There are some implementations of unsupervised learning which exploit the temporal link between two successive images. As an example, in Wang and Gupta (2015), tracked patches in a video stream are constrained to have similar internal representations. In Vondrick et al. (2015), the built representations are used to predict future states, whereas in Walker et al. (2016) they are used to define a probability over the trajectories of pixels in an image.

Other approaches to the problem of artificial perception claim that in order to build a truly perceiving agent, it is essential to take its actions into account. Instead of exploiting a mere sensory flow, the actions performed by the agent are processed in parallel. These approaches have been gathered under the term *Interactive Perception* (Bohg et al., 2016). Namely, the actions are used to learn representations consistent with ego-motion in Jayaraman and Grauman (2015), or to predict ego-motion from two successive images in Agrawal et al. (2015). In Oh et al. (2015) representations that allow the prediction of the next image conditioned on the agent's action are learned, while the effect of a physical action on an object is learned in Pinto et al. (2016). Both motor and sensory information have also been considered to build state representations consistent with robotic priors in Jonschkowski and Brock (2015).

Our approach is in line with these three paradigms: we process the temporal information between sensory inputs and the interaction of an agent with its environment, through unsupervised learning and a drive for prediction. Compared to the works previously cited, the specificity of ours is that we focus on the identification of spatio-temporally invariant structures from the sensorimotor flow of the agent.

Learning sensorimotor transition triplets $(\mathbf{s}_t, \mathbf{m}_t, \mathbf{s}_{t+1})$ share some similarity with learning $(\textit{object}, \textit{action}, \textit{effect})$ triplets in the affordance learning literature (Montesano et al., 2008; Zech et al., 2017), but these triplets are learned based on lower level modules extracting independent visual features for objects, effects, and eventually actions. In that respect, our positioning is more radical than most works in this literature, since we do not call upon such low-level feature extraction.

Other attempts have also been made to propose a computational model allowing for a sensorimotor grounding of knowledge for an artificial agent. One of the closest works with respect to ours in terms of investigating the nature of perception is Hay et al. (2018). In this work, the authors try to demonstrate how a naive agent may extract useful concepts from its sensorimotor experience. However, their concept learning framework assumes that there exists a separate reward function for each concept, an assumption that we consider too strong. In former works investigating the sensorimotor grounding of knowledge, such as Dorigo and Colombetti (1994) and Scheier and Pfeifer (1995), an external reinforcement signal was also used. In Cohen et al. (1997), a large amount of semantics is associated a priori with the actions performed by the agent and with its sensory stimulation, putting this work at a different level of abstraction. Finally, in Der et al. (1999) an agent uses a model of its sensorimotor interaction with the world in order to optimize at the same time its own structure (the parameters of the body of the agent) and the model itself . However, this work does not propose a mechanism to process the sensorimotor flow of the agent in order to build more abstract knowledge, like our agent does when learning to identify proto-objects as subgraphs in its general sensorimotor experience. In Maye and Engel (2011), an agent learns to predict the effect of its actions on its sensorimotor flow, depending on previous actions and states, learning a model which is very similar to ours. However, while the agent can learn by random exploration, the experimental setup contains no randomness, and the possible actions performed by the agent and its sensory inputs are defined at a more abstract level than ours. Importantly, clustering together sensory states to identify proto-objects is absent from these works, and robustness to randomness in the environment is not studied.

4.3. Limitations and Future Work

Despite its versatility, the approach we presented in this paper also suffers from multiple limitations. As revealed in the experiments, our algorithm is not able to handle the case where proto-objects are smaller than the receptive field. In the real world, however, proto-objects appear smaller to us than our field of view. As a consequence, instead of a single receptive field, several elementary receptive fields could be used in combination to define a visual field, as is the case in our own visual system. This should also open possibilities to tackle the problem of distinguishing multiple instances of the same proto-objects, and to reduce the ambiguity of a visual scene. Some preliminary results in this direction have already been published (Laflaquière, 2016). Instead of considering a small sensor moving in the environment, one could also imagine having a larger sensor with an attention mechanism focusing on a small part of it. Besides, the implementation presented here was intended to illustrate fundamental mechanisms making it possible to extract proto-objects, but would not scale to a more realistic setting. In a real-life context, the quantification of the sensorimotor experience of the agent would need a way larger amount of memory and computation time, making the method intractable. A more relevant way to process the sensorimotor experience might require an algorithm able to directly process the sensorimotor data without a preliminary K-MEANS clustering stage. It should be rather clear from section 4.1 that combining some properties from the standard computer vision paradigm with ours is the way to go in order to address the discovery of objects in real world environments. As an immediate example, using a neural network taking the sensory states and motor commands as input, and predicting the next sensory input could be a promising alternative to the initial K-MEANS clustering stage. However, given their very different nature and underlying assumptions, combining both paradigms into a more general framework is a difficult problem which will require careful examination in the future. Finally, learning a more compact representation of the sensorimotor experience, with a tool such as a deep neural network instead of a graph might make it possible to compare our approach to common benchmarks used in the computer vision community.

Finally, when the complexity of the problem increases, or in order to deal with locally ambiguous sensory input, a hierarchical processing of the experience might be necessary. On the one hand, it has been shown that the hierarchical processing of information is probably one of the reasons of the success of deep networks (Bengio et al., 2013; Lin et al., 2017). On the other hand, from a more biological point of view, it has been shown that biological brains are organized hierarchically (Modha and Singh, 2010), while the interpretation of the reasons for a hierarchical processing have been investigated but are still subject to debate (Damasio, 1989; Fuster, 2006). A proto-object or even an object could then be detected through a hierarchy of features. This approach should also be followed to tackle the problem of ambiguity, by exploiting sequences of transitions in order to define contexts, instead of exclusively exploiting instantaneous transitions (Hemion, 2017).

AUTHOR CONTRIBUTIONS

NLH implemented the model, designed and performed the experiments and wrote the paper. AL and OS designed the experiments and wrote the paper.

FUNDING

This work was supported by the Association Nationale Recherche Technologie (ANRT) through a CIFRE convention (contract No 2016/0946).

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/frobt.2018.00070/full#supplementary-material

REFERENCES

Agrawal, P., Carreira, J., and Malik, J. (2015). "Learning to see by moving," in *Proceedings of the IEEE International Conference on Computer Vision*, Vol. 2015 Inter (Santiago), 37–45.

Bengio, Y., Courville, A., and Vincent, P. (2013). Representation learning: a review and new perspectives. *Pattern Anal. Mach. Intell. IEEE Trans.* 35, 1798–1828. doi: 10.1109/TPAMI.2013.50

Bohg, J., Hausman, K., Sankaran, B., Brock, O., Kragic, D., Schaal, S., et al. (2016). Interactive perception: leveraging action in perception and perception in action. *arXiv preprint arXiv:1604.03670*, 1–18.

Clark, A. (2013). Whatever next? Predictive brains, situated agents, and the future of cognitive science. *Behav. Brain Sci.* 36, 181–204. doi: 10.1017/S0140525X12000477

Cohen, P. R., Atkin, M. S., Oates, T., and Beal, C. R. (1997). "NEO: learning conceptual knowledge by sensorimotor interaction with an environment," in *Proceedings of the First International Conference on* (Miami, FL: IEEE), 248–255.

Damasio, A. R. (1989). Time-locked multiregional retroactivation: a systems- level proposal for the neural substrates of recall and recognition. *Cognition* 33, 25–62.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). "Imagenet: a large-scale hierarchical image database," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, 248–255. IEEE. Miami, FL

Der, R., Steinmetz, U., and Pasemann, F. (1999). Homeokinesis-a new principle to back up evolution with learning. *Comput. Intell. Modell. Control Autom.* 55, 43–47.

Doersch, C., Gupta, A., and Efros, A. A. (2015). Unsupervised visual representation learning by context prediction. *arXiv preprint arXiv:arXiv:1505.05192v3*, 1422–1430.

Dorigo, M., and Colombetti, M. (1994). Robot shaping: developing autonomous agents through learning. *Artif. Intell.* 71, 321–370.

Erhan, D., Bengio, Y., Courville, A., Manzagol, P.-A., Vincent, P., and Bengio, S. (2010). Why does unsupervised pre-training help deep learning? *J. Mach. Learn. Res.* 11, 625–660.

Friston, K., Kilner, J., and Harrison, L. (2006). A free energy principle for the brain. *J. Physiol. Paris* 100, 70–87. doi: 10.1016/j.jphysparis.2006.10.001

Fuster, J. M. (2006). The cognit: a network model of cortical representation. *Int. J. Psychopsychol.* 60, 125–132. doi: 10.1016/j.ijpsycho.2005.12.015

Gibson, J. J. (1979). *The Ecological Approach to Visual Perception*. Boston, MA: Houghton Mifflin.

Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., et al. (2015). Recent advances in convolutional neural networks. *arXiv preprint arXiv:1512.07108v6*, 1–14.

Hay, N., Stark, M., Schlegel, A., Wendelken, C., Park, D., Purdy, E., et al. (2018). "Behavior is everything–towards representing concepts with sensorimotor contingencies," in *AAAI*.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). "Deep Residual Learning for Image Recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 770–778.

Helmholtz, H. (1896). Handbuch der physiologischen Optik. *Monatshefte Mathematik Physik* 7, A60–A61.

Hemion, N. J. (2017). "Context discovery for model learning in partially observable environments," in *2016 Joint IEEE International Conference on Development and Learning and Epigenetic Robotics, ICDL-EpiRob 2016*, 294–299.

Hoffman, D. D. (2015). The interface theory of perception: natural selection drives true perception to swift extinction. *Sci. Rep.* 24, 1–1. doi: 10.1017/CBO9780511635465.009

Jain, A. K., Duin, R. P. W., and Mao, J. (2000). Statistical pattern recognition: a review. *IEEE Trans. Pattern Anal. Mach. Intell.* 22, 4–37. doi: 10.1109/34.824819

Jayaraman, D., and Grauman, K. (2015). "Learning image representations equivariant to ego-motion," *IEEE International Conference on Computer Vision* (Iccv), 1413–1421.

Jonschkowski, R., and Brock, O. (2015). Learning state representations with robotic priors. *Auton. Rob.* 39, 407–428. doi: 10.1007/s10514-015-9459-7

Laflaquière, A. (2016). Autonomous grounding of visual field experience through sensorimotor prediction. *arXiv preprint arXiv:1608.01127*.

Laflaquière, A., and Hemion, N. (2015). "Grounding object perception in a naive agent's sensorimotor experience," in *5th Joint International Conference on Development and Learning and Epigenetic Robotics, ICDL-EpiRob 2015* (Paris), 276–282.

Lin, H. W., Tegmark, M., and Rolnick, D. (2017). Why does deep and cheap learning work so well? *J. Stat. Phys.* 168, 1223–1247. doi: 10.1007/s10955-017-1836-5

Luxburg, U. V. (2006). A tutorial on spectral clustering a tutorial on spectral clustering. *Stat. Comput.* 17, 395–416. doi: 10.1007/s11222-007-9033-z

Maye, A., and Engel, A. K. (2011). "A discrete computational model of sensorimotor contingencies for object perception and control of behavior," *Proceedings - IEEE International Conference on Robotics and Automation* (Shanghai), 3810–3815.

Meila, M. (2015). "Spectral clustering: a tutorial for the 2010's," *Handbook of Cluster Analysis*, 753. Available online at: http://www.stat.washington.edu/mmp/Papers/ch2.2-arxiv.pdf

Modha, D. S., and Singh, R. (2010). Network architecture of the long-distance pathways in the macaque brain. *Proc. Natl. Acad. Sci. U.S.A.* 107, 13485–13490. doi: 10.1073/pnas.1008054107

Montesano, L., Lopes, M., Bernardino, A., and Santos-Victor, J. (2008). Learning object affordances: from sensory–motor coordination to imitation. *IEEE Trans. Rob.* 24, 15–26. doi: 10.1109/TRO.2007.914848

Oh, J., Guo, X., Lee, H., Lewis, R., Singh, S., and Arbor, A. (2015). Action-conditional video prediction using deep networks in Atari Games. *arXiv preprint arXiv:1507.08750*.

O'Regan, J. K., and Noë, A. (2001). What it is like to see: a sensorimotor theory of perceptual experience. *Synthese* 129, 79–103. doi: 10.1023/A:1012699224677

O'Regan, K., and Noe, A. (2001). A sensorimotor account of vision and visual consciousness. *Behav. Brain Sci.* 24, 939–1031.

Pinto, L., Gandhi, D., Han, Y., Park, Y.-l., and Gupta, A. (2016). The curious robot: learning visual representations via physical Interactions. *arXiv preprint arXiv:1604.01360v2*.

Rafael, E., Razi, A., Parr, T., Kirchhoff, M., and Friston, K. J. (2017). Biological self-organisation and Markov blankets. *bioRxiv*, 1–21.

Rao, R. P. N., and Ballard, D. H. (1999). Predictive coding in the visual cortex: a functional interpretation of some extra-classical receptive-field effects. *Nat. Neurosci.* 2, 79–87.

Rao, R. P. N., and Ballard, D. H. (2005). "Chapter 91: Probabilistic models of attention based on iconic representations and predictive coding," in *Neurobiology of Attention*, eds L. Itti, G. Rees, and J. K. Tsotsos (Burlington, NJ: Academic Press), 553–561. doi: 10.1016/B978-012375731-9/50095-1

Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. (2015). You only look once: unified, real-time object detection. *arXiv preprint arXiv:1506.02640v5*.

Scheier, C., and Pfeifer, R. (1995). "Classification as sensory-motor coordination a case study on autonomous agents," in *European Conference on Artificial Life*, 657–667.

Seth, A. K. (2014). A predictive processing theory of sensorimotor contingencies: Explaining the puzzle of perceptual presence and its absence in synesthesia. *Cogn. Neurosci.* 5, 97–118. doi: 10.1080/17588928.2013.877880

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., et al. (2013). Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*.

Vondrick, C., Torralba, A., and Pirsiavash, H. (2015). Anticipating visual representations from unlabeled video. *arXiv preprint arXiv:1504.08023v2*.

Walker, J., Doersch, C., Gupta, A., and Hebert, M. (2016). "An uncertain future: Forecasting from static images using variational autoencoders," in *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Vol. 9911 LNCS, 835–851.

Wang, X., and Gupta, A. (2015). Unsupervised learning of visual representations using videos. *arXiv preprint arXiv:1505.00687v2*.

Zech, P., Haller, S., Lakani, S. R., Ridge, B., Ugur, E., and Piater, J. (2017). Computational models of affordance in robotics: a taxonomy and systematic classification. *Adapt. Behav.* 25, 235–271. doi: 10.1177/1059712317726357

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