Learning Sensorimotor Concepts Without Reinforcement

Yasser Mohammad[1,2] **and Toyoaki Nishida**[1]
[1] Dept. of Intelligence Science and Technology, Kyoto University, Japan
[2] Electrical Engineering Dept., Assiut University, Egypt
yaserm@aun.edu.eg nishida@i.kyoto-u.ac.jp

Abstract

Agents engaged in lifelong learning can benefit from the ability to acquire new concepts from continuous interaction with objects in their environments which is a ubiquitous ability in humans. This paper advocates the use of sensorimotor concepts that combine perceptual and actuation patterns. Related representations to sensorimotor concepts are Predictive State Representation in dynamical systems, Affordance Based Concepts in language and Skills in reinforcement learning. The paper proposes a system for learning generalized sensorimotor concepts from unsegmented interactions between the agent and the objects in its environment that works in continuous action and observation spaces and in the same time require no reinforcement signals. A proof-of-concept experiment with the proposed system on a simulated e-puck robot is reported to support the applicability of the proposed approach.

An important capability for a lifelong learner is learning new concepts from its own interaction with different objects. Concept learning has a long tradition in AI research. In its simplest form, concept learning is a special classification problem in which the learner is given some examples of the concept to be learned and is expected to generalize this knowledge to new stimuli (Tenenbaum 1999). In this paper, we are interested in a more challenging concept learning task in which the agent is trying to build *concepts* based on its continuous unsegmented interactions with the world. The following section orients the proposed system within existing research in concept learning.

Related Work

Research in concept learning usually focuses on learning patterns either in the perceptual space of the agent or in its action space. Perceptual concepts correspond to objects, and environmental features. Action concepts, on the other hand, correspond to motion patterns that can be executed to achieve specific goals. Most existing research is conducted within a framework of reinforcement learning and in discrete action and perceptual spaces.

An example of perceptual concept learning can be found in (Xi et al. 2007) in which recurrent shape patterns (called motifs) are mined from shape databases. A recent example of learning concepts in the action space (basic behaviors) was reported in (Mohammad and Nishida 2012) and (Mohammad and Nishida 2010) . In this system, an agent (the imitator) receives a record of another agent's (the imitatee) interaction with the environment and uses a motif discovery algorithm to learn a discrete set of actions that represent recurring patterns in the action space of the imitatee. A limitation of this approach is that learned behaviors have nothing to do with the objects in the environment and for this reason they can be used to learn only patterns of behavior that do not involve any form of object manipulation. Another example of action space concept learning can be found in (Hajimirsadeghi 2010). In this case, concepts are learned incrementally in the form of HMMs but again they are limited to spatio-temporal concepts and cannot be used for learning object manipulation concepts.

The proposed system differs from all of these systems in that it learns generalized sensorimotor concepts that represent aspects of both the perceptual and action space of the agent. These concepts are learned from continuous interaction with the environment and provide a richer representation that can be utilized for action generation, planning and perceptual recognition. Gornaik argues convincingly that the dichotomy between perceptual concepts and action concepts does not allow for genuine semantic understanding to emerge (Gorniak 2005). We believe that this analysis is not limited to linguistic processing. For an agent to engage in meaningful grounded learning, it is important to relate its concepts to both its perceptual abilities and action repertoire. This is why, in this paper, we are more interested in generalized sensorimotor concepts.

The idea of using generalized sensorimotor concepts as a representational tool is not new. One proposed approach was predictive state representation in dynamical system modeling. Littman et al. use multi-step action-conditional predictions of *future* observations (called tests) to represent the state of an agent in a discrete-actions discrete-observations environment. This approach was proven to provide generality and compactness at least similar to model-based generative models (e.g. POMDP) while being grounded in data as much as history-based systems (e.g. K-order Markov models) (Littman, Sutton, and Singh 2002). Singh et al. showed that PSRs are more general than POMPDs and K-order

Markov Models (Singh, James, and Rudary 2004). One limitation of this approach is the discrete actions/observations assumption. Affordance-based concepts (ABCs) are related to sensorimotor concepts in that they represent objects not by their perceptual properties but by the *affordances* they provide (e.g. possible *future* interactions with them). Gorniak used ABCs in a linguistic context to learn concepts related to a rule-playing game (Gorniak 2005). Because concepts are not purely subjective while not purely objective, modeling them with *perceived affordances* that are grounded in both perception and action can provide a solution to the subject-object gab in mental representations (Roy 2005). The work presented here, is not concerned with the linguistic aspect of this approach but utilizes affordances for concept abstraction based on a tight coupling between perception and action.

Learning sensorimotor concepts is also not new. In (Klingspor and Morik 1995), a system for learning operational concepts was proposed. The system uses a set of predefined functions and their parameter ranges to learn a set of features that are used for concept learning. Concepts in this system are higher order sensorimotor concepts that represent concrete operations. Learning adequate features require several evaluations of concept *quality* which is always related to a specific task. The proposed system does not require these predefined functions and does not need any evaluations of concept *quality*. Constructing Skill Trees (CST) is a form of sensorimotor concept learning in the context of reinforcement learning. In this system, a change point discovery algorithm is used to segment demonstrations into different *skills*. Each skill has a goal and is assigned a suitable set of sensorimotor dimensions to work with (abstractions) (Konidaris et al. 2011). Skills are very useful concepts that can tremendously reduce the complexity of reinforcement learning problems. Nevertheless, a reinforcement signal is necessary for this system to work. A somewhat different approach within the hierarchical reinforcement learning approach is Self-Organizing Distinctive State Abstraction (SODA) (Provost, Kuipers, and Miikkulainen 2007) which learns first sensory representations during a babbling phase then learns two strategies to move between them.

Most of these approaches assume a discrete-actions discrete-observations context and/or require a reinforcement signal for learning. The work presented in this paper differs from these systems in working in continuous perceptual and action spaces and requiring no reinforcement. Learning in continuous sensorimotor spaces has already been studied. For example, Pierce and Kuipers developed a system for map learning with uninterpreted sensors and effectors that assume an almost everywhere approximately linear sensorimotor apparatus (Pierce and Kuipers 1997). The proposed system does not require this assumption and it allows an agent to learn from observing the interaction between another agent and the environment or from analyzing its own exploration without the need for preset goals.

Consider a robot living in a world populated by different objects as seen in Fig. 1. The robot starts with just a drive to move around and avoid colliding with anything. In this scenario, the robot has no specific goal and no reinforcement is given. Nevertheless, given extended interaction with the environment, the robot should be able to gradually abstract its sensorimotor patterns into concepts that can be then used for both action generation in a feedback loop and for active object recognition. For example, once the robot goes around several cylinders and cubes, it learns that there is some common sensorimotor patterns that are of importance at least because of their recurrence. Nevertheless, these patterns can be divided into two clusters. One corresponds with cylinders and the other with cubes. These sensorimotor patterns allow the robot after that to go around similar objects by activating the concept (emulation). It also allows it to discover the *type* of a newly encountered object by activating each concept in turn and comparing its perceptual information with the stored information in the concept (recognition). This also allows the robot to generate new concepts for newly encountered object that combine these learned concepts. Possible ways to utilize learned concepts will be discussed in details in the following sections after discussing the learning algorithm.

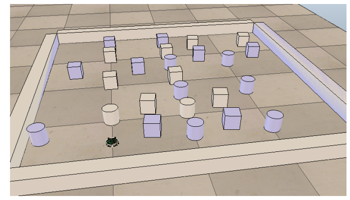

Figure 1: e-puck in its simulated environment.

This form of concept learning is not alien to human experience. Consider children's interaction with their environment. Piaget maintains that there are instinctual behavioral schemata (e.g. suckling) that are refined through assimilation and accommodation processes with no need of external reinforcement leading to the formation of new behavioral and operational schemata that allow the child to develop its conception of the world, its objects, and its relations (Piaget 1953). The proposed form of concept learning works in a similar way by allowing the agent to autonomously discover salient features of its experience (in both the perceptual and action spaces) and abstracting these features into refinable reusable units (a process similar to the accommodation process). It allows that agent to then perceive newly encountered experiences in terms of its learned concepts (a process similar to the assimilation process).

The system does not have separate learning and testing phases because concepts are incrementally learned as long as the agent is active and they are activated whenever the current interaction is similar to the ones used to learn them. Nevertheless, we will separately discuss concept learning and concept utilization. The rest of this paper describes how concepts are learned, and how they are utilized after learning. The system is then evaluated using the scenario described in this section (Fig. 1).

Learning Concepts

The following definitions will be used throughout this paper:

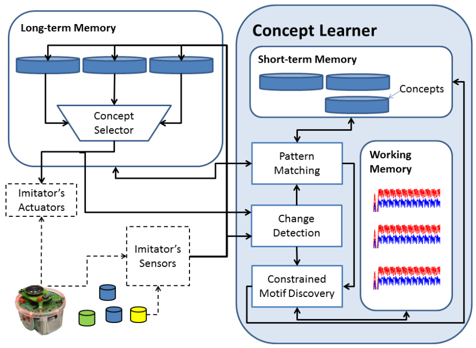

Figure 2: Overview of the proposed concept learner. Short-term and Long-term memories contain concepts learned from the interaction while the working memory contains candidates for concept membership.

Time Series An ordered list of real valued n dimensional vectors where n is the *dimensionality* of the time series and T is its length.

Sensorimotor Signal A *time series* created by adding synchronized sensor readings and actuator commands to form a sensorimotor vector at each timestep. All values are considered real and assuming that the robot has n_s sensed values and n_m actuation dimensions, the dimensionality of its sensorimotor signal is $n_s + n_m$.

Point-wise Distance Given two time series x and y of the same length L, their pair-wise distance is an unordered list of cardinality L where $d(t)$=$[x(t) - y(t)]^2$.

Concept A concept is a tuple $\{c, \mu, \sigma, n\}$ where c is a time series of dimensionality $n_s + n_m$ representing the mean of a set of instances of this concept, μ is the mean of point-wise distances between these instances (e.g.$\mu = \frac{1}{n(n-1)} \sum_{i=1}^{n} \sum_{j=i+1}^{n} \frac{1}{L} \sum [c_i(t) - c_j(t)]^2$), σ is the variance of point-wise distances between these instances, and n is the weight of this concept defined as the number of instances corresponding to it. A *concept instance* is a sensorimotor signal that was used to learn it.

In this paper we provide an approximate solution of concept learning. Concept learning is an instance of a more general problem called motif discovery in data mining. In motif discovery, the input is a time series and the output is a set of motifs corresponding to recurring patterns in this time series. An exact solution to motif discovery in real-valued time series (assuming a metric distance measure) was proposed in (Mueen et al. 2009). This solution is not incremental and this limits its utility for lifelong learning. An incremental extension was proposed in (Mueen and Keogh 2010) that works with multidimensional data (similar to concepts) but this solution can learn a single motifs only if two instances of it are within the working memory of the agent. A recent algorithms was proposed to find k top motifs but still these motifs are always two-instances each and the two instances must be in the working memory of the agent to be discovered (Lam, Pham, and Calders 2011). The proposed algorithm tries to extend the utilization of working memory by concentrating the search around points at which a change of dynamics is estimated. This allows the agent to discover concepts that are further apart without extending its working memory. It also uses an incremental merge process to generate concepts of more than two instances each. This is useful for later modeling of concepts (e.g. using HMMs) and reduces the sensitivity to bursts of systematic distortions common in real world sensing situations.

Algorithm 1 shows the general operation of the proposed learner and its architecture is shown in Fig. 2. It has three memory levels (working memory (W) and short term memory (SM) used to learn concepts and long term memory (LM) used to store and utilize them later). It keeps a fixed-length buffer of the incoming sensorimotor signal (sm) that is continuously searched for points at which the generating dynamics of each dimension is likely to be changing (change points) (*cpd()* call). At change points, the learner first consults the long-term memory to find if the sensorimotor patterns around the change point can be assimilated into one of the learned concepts (first *matchPattern()* call). If so, the concept is accommodated to refine its representation and the corresponding concept is activated. Otherwise, the learner tries matching concepts in the short term memory (second *matchPattern()* call). If all fails, the learner passes the contents of the working memory after adding the current buffer to it (replacing oldest entry if no space is available) to an incremental motif discovery algorithm that compares it with contents of the short-term memory trying to find any other sensorimotor signal that contains a similar pattern. If so, a new concept is created and added to the short-term memory.

Algorithm 1 Concept Learner Algorithm

procedure CL(L_{min},L_{max},n_{min},LM,SM,W,sm)
  for each sensorimotor vector sm_t **do**
    if cpd(sm,t,M,$L_{max} - M$) **then**
      $s \leftarrow sm\,(t - L_{max} - 1 : t + L_{max})$
      if C_k ←matchPattern(s,LM) **then**
        Update matched concept (C_k)
      else
        if C_k ←matchPattern(s,SM) **then**
          Update matched concept (C_k)
          if $C_k\,[n] > n_{min}$ **then**
            $LM \leftarrow LM \bigcup C_k$
            $SM \leftarrow SM - C_k$
        else
          CMD(W,s,L_{min},SM,α,ζ)
          if W is full AND CMD failed **then**
            Remove oldest member of W
          if CMD failed **then**
            $W \leftarrow W \bigcup s$

The following subsections describe each component of this system.

Incremental Change Point Discovery

Change Point Discovery (CPD) is defined as follows: Given a time series $x(t)$ of length T, find another time series of the same length $(c(t))$ where $0 \le c(t) \le 1$ and higher $c(t)$ represents a higher estimate that there is a *change in generating dynamics* in x near the timestep t.

We base our change point discovery on a variant of Singular Spectrum Analysis change point discovery called the Robust Singular Spectrum Transform (RSST) (Mohammad and Nishida 2011).

The goal of the $RSST$ transform is to find for every point $x(i)$ the difference between a representation of the dynamics of the few points before it (i.e. $x(i-p) : x(i)$) and the few points after it with possibly few skipped points (i.e. $x(i+g) : x(i+g+f)$). This difference is normalized to have a value between zero and one and named $x_s(i)$.

The dynamics of the points before the current point are also represented using the Hankel matrix (B) which is calculated as:

$$B(t) = [seq(t-n), ..., seq(t-1)] \tag{1}$$

where $seq(t) = \{x(t-w+1), ..., x(t)\}^T$.

The dynamics of the points after the current point are also represented using the Hankel matrix (A) which is calculated as:

$$A(t) = [r(t+1), ..., r(t+1+n-1)] \tag{2}$$

where $r(t+i) = \{x(t+i), ..., x(t+i+w-1)\}^T$

Once more, we apply SVD (Singular Value Decomposition) to both A and B and keep only the top most $l_A(t)$ and $l_B(t)$ Eigen vectors (calculated using the largest-gab in Eigen values). This gives us two more subspaces ($U_{l_A}(t)$ and $U_{l_B}(t)$). The change score is then calculated by:

$$c(t) = 1 - \cos(\theta) \tag{3}$$

where θ is the angle that this vector have with the subspace spanned by $U_{l_A}(t)$ and $U_{l_B}(t)$.

To decide that a change is likely to be happening in the generating dynamics, we aggregate RSST's output with a small window of size w_c and once this value exceeds a predefined threshold T_c a change is announced and both the pattern matching and the constrained motif discovery modules are activated. Because the size of the Hankel matrices is fixed, this component requires constant space and time for its execution per point. This algorithm is run over all dimensions of the sensorimotor signal perceived by the robot (its sensor readings and actuator commands) and a change is announced if the threshold is exceeded in ANY dimension.

Pattern Matching

Once the change detector announces a change, pattern matching is started. The pattern matcher is a classifier that receives the sensorimotor signal (sm) of length $2L_{max}$ centered at the change point and compares it to all concepts in the long and short term memories (in that order). If a distance limit is specified (d_{max}) then it can be used and a match is announced with concept i, iff $D(sm, c_i) < d_{max}$ where c_i is the member c of the concept i (see concept definition). This approach has two major problems. Firstly, it is hard to specify a sensible value for d_{max}, and secondly, it is sensitive to outliers as few outliers can drive the distance large enough to miss a match.

In this paper we use a statistical test to drive the matching process. The point-wise distance $d(sm_j, c_i)$ is calculated between every subsequence of sm of the length $|c_i|$ and every long-term memory concept C_i. These distances are then compared with stored mean and variance (m_i,v_i) of the concept using a single-sided t-test. If the null-hypothesis (measured mean is less or equal stored mean) was rejected for only one concept, a match is announced.

When a match is announced with concept k, the matching subsequence sm_j is added to the motif by updating concept members as follows:

$$\bar{\mu} = \frac{1}{|c_k|} \sum d(sm_j, c_k) \tag{4}$$

$$\bar{\sigma} = \frac{1}{|c_k|} \sum (d(sm_j, c_k) - \bar{\mu}) \tag{5}$$

$$\bar{\mu}_k = \mu_k \tag{6}$$

$$\mu_k = (n_k \times \mu_k + \bar{\mu}) / (n_k + 1) \tag{7}$$

$$\sigma_k = \frac{1}{n_k+1} (n_k \times \sigma_k + (\bar{\mu} - \bar{\mu}_k) \times (\bar{\mu} - \mu_k)) \tag{8}$$

$$c_k = \frac{1}{n_k+1} (n_k \times c_k + sm_j) \tag{9}$$

$$n_k = n_k + 1 \tag{10}$$

Incremental Constrained Motif Discovery

When the change detector announces a change and the pattern matcher fails to match it to any existing concept, the constrained motif discovery module is activated. The fact that this module is not activated except at change points reduces the required working memory size dramatically as most of the sensorimotor signal is ignored. This module tries to discover motifs of variable length within the range ($L_{min} : L_{max}$) within the working memory of the agent and move them to the short-term memory. The new sensorimotor signal (sm) of length $2L_{max}$ centered at the change point is used as a candidate window for a new concept (w_{new}). After completing CMD, if no motifs are found and the working memory is full, its oldest sensorimotor signal is removed and sm is added to it.

The proposed algorithm is based on the algorithm proposed by Catalano et al. (Catalano, Armstrong, and Oates 2006) but it differs from it in requiring no random windows from the data. This makes the algorithm fully incremental. The algorithm has one parameter $\hat{w}$ which must be less than L_{min} (we set $\hat{w} = L_{min}/5$ for all our experiments). The steps of the algorithm are shown in Algorithm 2. The function $sub(x, \hat{w})$ finds all subsequences of x of length $\hat{w}$. The algorithm compares the new sensorimotor signal with an existing one by finds all distances between their subsequences of length $\hat{w}$. We extend the $\hat{w}$ subsequences that gave minimum distance as long as the distance between resulting longer subsequences is below ζ of the difference between maximum and minimum distances. We accept that the

Algorithm 2 Incremental Motif Discovery Algorithm

procedure CMD(W,w_{new},L_{min},SM, α,ζ)
  $\hat{w} \leftarrow \alpha \times l_{\min}$
  for $w \in W$ **do**
    for Every dimension d of w **do**
      $Motif_d \leftarrow \phi$
      $mx \leftarrow \arg\max \left(d \left(sub(w_{new}, \hat{w}), sub(w, \hat{w})\right)\right)$
      $mn \leftarrow \arg\min \left(d \left(sub(w_{new}, \hat{w}), sub(w, \hat{w})\right)\right)$
      $d_x \leftarrow \max \left(d \left(sub(w_{new}, \hat{w}), sub(w, \hat{w})\right)\right)$
      $d_n \leftarrow \min \left(d \left(sub(w_{new}, \hat{w}), sub(w, \hat{w})\right)\right)$
      $xs \leftarrow w_{new} \left((mn - \hat{w} + 1 : mn)\right)$
      $mx \leftarrow$ corresponding subsequence of w
      while $d\left(xs, mx\right) < \zeta \left(d_x - d_n\right) + d_n$ **do**
        Extend xs and mx
      if $|xs| \geq l_{min}$ **then**
        $c \leftarrow mean\left(xs, mx\right)$
        $\mu \leftarrow mean(d(xs, mx))$
        $\sigma \leftarrow variance(d(xs, mx))$
        $Motif_d \leftarrow \{c, \mu, \sigma, 2\}$
      $Motif \leftarrow combine(Motif_1, .., M_{n_s+n_m})$
      if Sensorimotor Motif was found **then**
        $SM \leftarrow SM \bigcup Motif$

two subwindows are similar if at least two subsequences of length l_{min} were similar according to the aforementioned test. Because we are sure that both window lengths is greater than L_{max} we expect that at least some of the distances will be higher than any distances between corresponding subsequences of the motif (if one exists) and this gives us a natural limit on the allowable distance.

CMD finds motifs in a single dimension of the sensorimotor signal. Motifs found at different dimensions are then ANDed. This means that if motifs were found in the same time frame in multiple dimensions, they will be combined to form a single multidimensional motif. This results in three types of concepts that are getting added to the short-term (and later long-term) memory of the agent: Perceptual motifs in which all the dimensions are from the perceptual dimensions, Motor motifs in which all the dimensions are from actuator dimensions, and Sensorimotor concepts that combine both.

Perceptual concepts represent relations to the objects in the environment that do not depend on the behavior of the agent. This means that they are of no value for future interactions and for this reason we ignore them. Motor concepts are more interesting and provide a model of recurrent activities that are executed by the agent but that do not change its relations to the objects in the environment. This means that these motifs represent patterns of behavior that achieve a modification of the global state of the agent in the environment (e.g. motion in an empty arena). This makes motor motifs important for cases of reinforcement learning (because of the inherent state representation) and imitation learning (see for example (Mohammad and Nishida 2012)). Nevertheless, they provide no information about the objects in the environment and for this reason we ignore them in this paper. Sensorimotor motifs represent concrete interactions between the agent and its environment and they are stored in short-term memory as sensorimotor concepts once found.

Higher order concepts that are composed from activation patterns of learned concepts can also be learned using a standard motif discovery algorithm (e.g. PROJECTIONS (Buhler and Tompa 2002)). This is not explored further in this paper due to lack of space.

Proof-of-Concept Experiment

This section reports a proof-of-concept experiment on the scenario shown in Fig. 1. Evaluation was done using V-REP's (V-REP) simulator of the e-puck robot (EPFL) built on the ODE physics simulator. The robot lives in a 2m by 2m environment populated by two types of objects (cylinders and cubes). It has eight proximity sensors that have a maximum range of 5cm and two motors (left and right) that control the speed of left and right wheels. The maximum rotational velocity of both motors lead to 1 rpm which is called ν_{max}. These were the only sensors and actuators used in this experiment leading to a sensorimotor signal dimensionality of 11. The software was implemented using a reactive control architecture called L_0EICA (Embodied Interactive Control Architecture) (Mohammad and Nishida 2008).

Robot software consists of an open ended set of processes (behaviors). Each behavior is connected to both sensors and actuators and has two basic control ports: *enable* which determines whether or not it is running. *activation level* which determines the weight of its actuation commands. In this paper, we use a winner-gets-all action integration strategy that allows the single behavior having highest activation level to control the robot. Other behaviors that are enabled continue to execute but are not allowed access to robot actuators.

The robot had initially three behaviors. The first (called $Wander$) just sends ν_{max} to both motors making the robot charge forward but every 1 minute it executes a rotation by replacing randomly one of the two motor speeds by $-\nu_{max}$ for a random amount of time between one and 10 seconds. This behavior has an activation level of 0.1 and it is always enabled.

The second behavior (called $Avoid$) implements collision avoidance which has an activation level of one making it the winner in any competition .

The third and final initial behavior is the concept learner detailed in this paper. Once a concept is learned and moved to the long-term memory, a behavior is created to associate with the concept. The behavior continuously sends the motor commands stored in the concept (c) and measures the difference between the stored and actual sensor readings for all the dimensions participating in the concept. The activation level of the concept is self-calculated as: $a = 1 - \frac{e^{\hat{d}}}{1+e^{\hat{d}}}$

where $\hat{d}$ is calculated as the point-wise distance between stored and perceived sensor values divided by the length of the concept executed so far. This means that the activation level of any concept goes up and down with its ability to predict the perceived signal. The activation level is always between 0.5 and zero.

The parameters of the simulation were: L_{min}=120 corresponding to 6 seconds of motion, $L_{max} = 10 \times L_{min}$,

$M = L_{min}/10$, $\zeta = 0.2$, $\alpha = \zeta$, $|W| = 10$, $|SM| = 2 \times |W|$,$|LM| = \infty$.

Fifty sessions were conducted to test the proposed system. At every session, 25 objects were placed randomly within the arena. Objects were not allowed to be less than 10cm apart and had higher probability to cluster in straight lines to provide a richer environment for the robot. A random number between 8 and 20 were cylinders and the rest were cubes. The concept learner was activated (by setting its enable port to one) only once every ten sessions. During the rest of the sessions, the robot only interacted with its learned concepts. During motif discovery, a constant valued signal was always ignored as trivial. Constant values in one of the motor dimensions were reintegrated in the final concepts to provide a complete actuation command.

The setup of this experiment generated four different concepts corresponding to cylinders, cubes, walls, and corners. In all of the five learning sessions, the agent was able to learn the four concepts. In some cases extra concepts corresponding to a concatenation of one or two of these behaviors were learned (1 extra concept was learned in two of the sessions). There was a total of 11 partial concepts learned in the five sessions (average 2.2 per session) that corresponded to a part of the four basic patterns. This shows that change point discovery was able to guide the search for recurrent patterns to the boundaries of concepts.

During the 45 evaluation sessions (in which concept learning was disabled), we compared the most active concept and the nearest object to the robot and found that recognition accuracy was 83.7% where most of the mistakes corresponded to confusing walls and partial cube concepts. A problem with the system was that partial concepts had activation layers higher than the corresponding full concept. This problem will be considered in future work.

Conclusion

This paper presents a new concept learning system that is able to learn sensorimotor concepts from agent's interaction with objects in its environment. The system needs no reinforcement and can be operated as a background process that allows the agent to autonomously build a set of discrete units for explaining its interactions. The system is incremental in the sense that it does not need to keep a record of the complete interaction with the environment. This is achieved by focusing the search for recurring patterns around change points in the sensorimotor signal using a change point discovery algorithm A proof of concept experiment was reported that shows the system in action. 83.7% of ground truth concepts were learned by the system without any external reinforcement signals.

References

Buhler, J., and Tompa, M. 2002. Finding motifs using random projections. *Journal of computational biology* 9(2):225–42.

Catalano, J.; Armstrong, T.; and Oates, T. 2006. Discovering patterns in real-valued time series. In *PKDD*, 462–469.

EPFL. http://www.e-puck.org.

Gorniak, P. 2005. *The affordance-based concept*. Ph.D. Dissertation, MIT.

Hajimirsadeghi, H. 2010. Conceptual Imitation Learning : An Application to Human-Robot Interaction. In *JMLR Workshop: 2nd Asian Conference on Machine Learning*, 331–346.

Klingspor, V., and Morik, K. 1995. Towards Concept Formation Grounded on Perception and Action of a Mobile Robot. In *the 4th Intern. Conference on Intelligent Autonomous Systems*, 271–278.

Konidaris, G.; Kuindersma, S.; Grupen, R.; and Barto, A. 2011. Robot learning from demonstration by constructing skill trees. *IJRR* 31(3):360–375.

Lam, H.; Pham, N.; and Calders, T. 2011. Online discovery of top-k similar motifs in time series data. In *SDM*, 1004–1015.

Littman, M.; Sutton, R.; and Singh, S. 2002. Predictive representations of state. *NIPS* 1555–1561.

Mohammad, Y. F. O., and Nishida, T. 2008. A cross-platform robotic architecture for autonomous interactive robots. In *IEA/AIE*, 108–117.

Mohammad, Y., and Nishida, T. 2010. Learning interaction protocols using augmented baysian networks applied to guided navigation. In *IEEE/RSJ IROS*, 4119–4126. IEEE.

Mohammad, Y., and Nishida, T. 2011. On comparing SSA-based change point discovery algorithms. In *2011 IEEE/SICE SII*, 938–945.

Mohammad, Y., and Nishida, T. 2012. Fluid imitation: Discovering what to imitate?. *IJSR*, 4(4): 369–382.

Mueen, A., and Keogh, E. 2010. Online discovery and maintenance of time series motifs. *KDD*, 1089–1098.

Mueen, A.; Keogh, E.; Zhu, Q.; and Cash, S. 2009. Exact discovery of time series motifs. *Proc. of SIAM SDM*, 1–12.

Piaget, J. 1953. *The origin of intelligence in the child*. Routledge & Kegan Paul.

Pierce, D., and Kuipers, B. 1997. Map learning with uninterpreted sensors and effectors. *Artificial Intelligence* (92):169–229.

Provost, J.; Kuipers, B. J.; and Miikkulainen, R. 2007. Self-organizing distinctive state abstraction using options. In *Proceedings of the 7th International Conference on Epigenetic Robotics*.

Roy, D. 2005. Grounding words in perception and action: computational insights. *Trends in cognitive sciences* 9(8):389–96.

Singh, S.; James, M.; and Rudary, M. 2004. Predictive state representations: A new theory for modeling dynamical systems. In *UAI*, 512–519.

Tenenbaum, J. B. 1999. *A Bayesian Framework for Concept Learning*. Ph.D. Dissertation, MIT.

V-REP. ttp://www.v-rep.eu/.

Xi, X.; Keogh, E.; Wei, L.; and Mafra-Neto, A. 2007. Finding motifs in databases of shapes. In *SIAM SDM*.