Bootstrap Learning of Foundational Representations*

Benjamin Kuipers, Patrick Beeson, Joseph Modayil and Jefferson Provost
Computer Science Department
University of Texas at Austin
Austin, Texas 78712 USA
{kuipers,pbeeson,modayil,jp}@cs.utexas.edu

Abstract

To be autonomous, intelligent robots must learn the foundations of commonsense knowledge from their own sensorimotor experience in the world. We describe four recent research results that show how a robot learning agent can bootstrap from the "blooming buzzing confusion" of the pixel level to a higher-level ontology including distinctive states, places, actions, and objects.

Introduction

Commonsense knowledge is a bottleneck problem on the way to artificial intelligence (McCarthy 1968). Common sense, and hence most other human knowledge, is built on knowledge of a few foundational domains, such as space, time, action, objects, causality, and so on (Piaget & Inhelder 1967; Minsky 1975). Spatial knowledge is arguably the most fundamental of these foundational domains (Lakoff & Johnson 1980). We are investigating how the foundations of spatial knowledge can be learned from unsupervised sensorimotor experience.

We have done extensive work on human and robot knowledge of large-scale space (the cognitive map), leading to the Spatial Semantic Hierarchy (Kuipers 2000; Remolina & Kuipers 2004). The multiple levels of the SSH demonstrate how higher levels of representation can be based on lower, simpler levels. The SSH Control level, the lowest, makes a good target for bootstrap learning.

The basic idea behind bootstrap learning is to reach a learning goal by composing multiple simple machine learning methods, using weak but general learning methods to create the prerequisites for applying stronger but more specific learning methods. The result is a lattice of learning methods that collectively learn the desired knowledge.

We assume that a learning agent[1], human or robot, starts with a low-level ontology for describing its sensorimotor interaction with the world. William James called this the "blooming buzzing confusion" that confronts the infant from its unfamiliar senses. From a robotics perspective, we call it the "pixel level", referring to the individual pixels of a camera image, the individual measurements in a laser range scan, the incremental motions caused by motor signals, the individual cells of an occupancy grid map, and so on. The learning task is to create useful higher-level representations for space, time, actions, objects, etc, to support effective planning and action in the world, bootstrapping up from experience at the pixel level.

As a matter of research strategy, we place secondary importance on determining whether a particular learning problem is solved by the species or by the individual. When comparing across species, it is clear that knowledge that is innate in one species is learned by individuals in another. We focus our attention on computational modeling of the learning process, and postpone the decision of where to place the evolutionary/developmental boundary. In general, we will write as though all learning is done by the individual learning agent, but this is not to preclude evolutionary learning.

Another matter of research strategy is that we do not begin with a commitment to a particular set of computational primitives. We assume that the learning agent has access to some collection of domain-independent statistical learning methods, but not to knowledge about the nature of its own sensors and effectors, or of the environment it lives in. Once we have a sufficient set of successful learning methods, we can begin identifying minimal subsets compatible with particular implementation technologies, including biological ones.

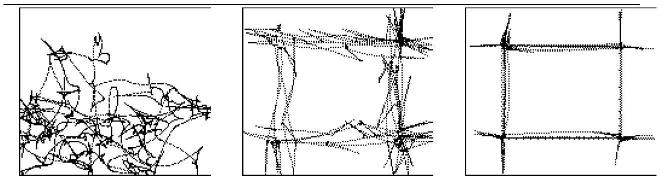
In this paper, we describe four recent research results that carry us significantly further toward the goal of autonomous learning of foundational representations for commonsense knowledge.

Learning from Uninterpreted Sensors and Effectors

The lowest level problem is faced by a learning agent in an unknown environment with unknown sensors and effectors.

*This work has taken place in the Intelligent Robotics Lab at the Artificial Intelligence Laboratory, The University of Texas at Austin. Research of the Intelligent Robotics lab is supported in part by the Texas Higher Education Coordinating Board, Advanced Technology Program (grant 003658-0656-2001), and by an IBM Faculty Research Award.

[1]We use the term "robot" to refer to the physical system and its sensors and effectors. The "learning agent" is the computational process observing and learning to control the robot. Body and mind, if you wish.



(a) (b) (c)

Figure 1: Exploring a simple world at three levels of competence. (a) The robot wanders randomly while learning a model of its sensorimotor apparatus. (b) The robot explores by randomly choosing applicable homing and open-loop path-following behaviors based on the static action model while learning the dynamic action model (see text). (c) The robot explores by randomly choosing applicable homing and closed-loop path-following behaviors based on the dynamic action model.

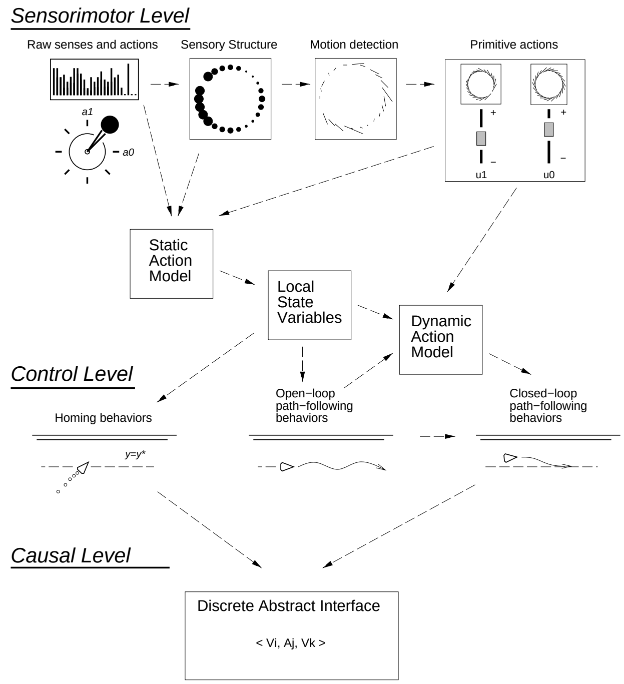


Figure 2: The lattice of learning methods and their results, from Pierce & Kuipers [1997].

Our goal is to learn the foundation for the Spatial Semantic Hierarchy (Kuipers 2000). The SSH rests on a set of hill-climbing and trajectory-following control laws and the knowledge of the sensorimotor interface to support them. How can this knowledge be learned from unsupervised experience?

Pierce and Kuipers (Pierce & Kuipers 1997) answered this question in the context of a simulated mobile robot with unknown sensors and effectors. The learning agent conducted a variety of experiments and analyzed the data, building a hierarchy of representations of both the sensory and motor systems, and eventually creating control laws that could define distinctive states (Figure 1). The analysis had the following steps.

(1) Gather observations during random sequences of actions. First, coarsely cluster the sensors according to the qualitative properties of a histogram of values returned by each sensor. Then, within appropriate clusters, compute pairwise correlations among sensor values and interpret them as similarity measures.

(2) Assign the sensors in a cluster to positions in a high-dimensional space reflecting their pairwise similarities. Project to a low-dimensional subspace (2D in our examples) that best accounts for most of the variance in the cluster. Once sensor values have a spatial as well as temporal dependence, we can calculate spatial as well as temporal derivatives, and thus define motion fields.

(3) The motion fields corresponding to different motor signals are analyzed using principal component analysis to determine the most significant motion effects and the motor signals that correspond to them. These signals are used as the natural primitives for the motor space.

(4) Higher-level sensory features are proposed, based on the spatial and temporal attributes of the field of primitive sensory values. These include features such as discontinuities, local minimum and local maximum, with magnitude, position, and scope. Proposed features are evaluated according to stability, predictive power, and extensibility.

(5) Evidence is collected of the effects of primitive motor commands on higher-level features, searching for motor commands that change features in predictable ways. "Local state variables" are defined for particular neighborhoods in the environment. Trajectory-following and hill-climbing control laws are defined according to which local state variables correspond to features that are both observable and controllable.

(6) Open-loop control laws are defined by identifying commands that reliably change one feature while keeping another one relatively constant. Closed-loop control laws are defined by searching for and identifying commands that can reduce deviations in the relatively constant feature, actively keeping it close to a desired setpoint. (Think of moving along a wall, turning slightly to maintain a desired distance from it. Compare figures 1(b,c).)

Figure 1 shows exploration traces at three stages of the learning process. The analysis uses a variety of mathematical methods, but only ones that can be applied to weakly interpreted data, using local computations such as neural networks. Figure 2 shows the lattice of learning methods that supported these conclusions.

One lesson from (Pierce & Kuipers 1997) is that learning even an apparently simple sensorimotor skill such as wall-following requires a large number of distinct learning algorithms, constructing a lattice of different representations of the sensory and motor capabilities of the robot.

Learning Distinctive States

In (Pierce & Kuipers 1997), the learning of high-level sensory features and hill-climbing and trajectory-following control laws made use of certain background knowledge, albeit of an abstract and domain-independent kind. In order to eliminate this use of background knowledge, (Provost,

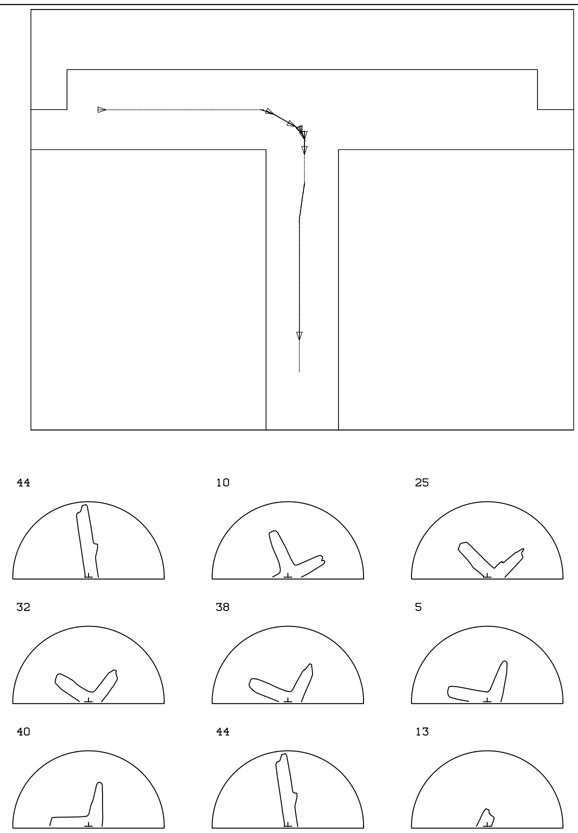


Figure 3: **Navigation using Learned Abstraction**. The upper diagram shows the robot's environment and an example episode after the agent has learned the task using the $\mathcal{A}^1$ actions. The triangles indicate the distinctive states the robot is in at the start of each $\mathcal{A}^1$ action. The bottom part of the figure shows the sequence of perceptual features corresponding to these distinctive states. The narrow line indicates the sequence of $\mathcal{A}^0$ actions used by the $\mathcal{A}^1$ actions. Navigating to the goal requires only 9 $\mathcal{A}^1$ actions, instead of hundreds of $\mathcal{A}^0$ actions – task diameter is vastly reduced.

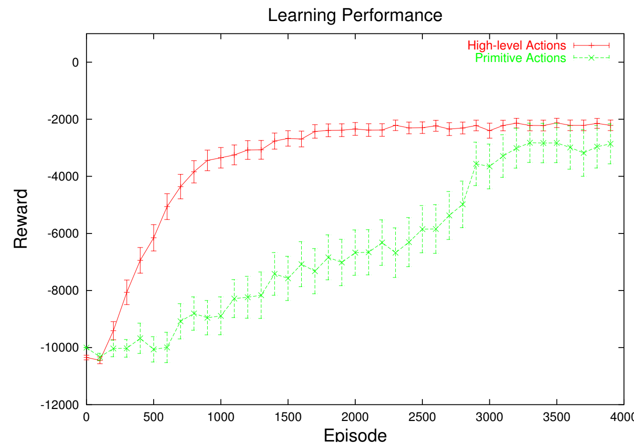


Figure 4: **Learning Performance** Comparison of the reward earned per episode using $\mathcal{A}^0$ actions vs. using $\mathcal{A}^1$ actions. Each curve is an average of 12 runs using each of 10 different learned feature sets. Error bars indicate +/- one standard error.

Kuipers, & Miikkulainen 2004) are revising our approach to use more generic learning methods such as self-organizing maps and reinforcement learning to achieve the same goals.

Modern robots are endowed with rich, high-dimensional sensory systems, providing measurements of a continuous environment. In addition, many important real-world robotic tasks have *high diameter*, that is, their solutions require a large number of primitive actions by the robot, for example, navigating to distant locations using primitive motor control commands. Reinforcement learning (RL) methods show promise for automatic learning of robot behavior, but extending these methods to high-dimensional, continuous, high-diameter problems remains a major challenge. Thus, the success of RL on real-world tasks still depends on human analysis of the robot, environment, and task to provide a useful set of perceptual features and an appropriate decomposition of the task into subtasks. Our goal is to create autonomous learning agents, relying on few assumptions about the nature of the robot and its world.

Self-Organizing Distinctive-state Abstraction (SODA) is a new method for automatic discovery of high-level perceptual features and large-scale actions for reinforcement learning in continuous environments (Provost, Kuipers, & Miikkulainen 2004). SODA requires little prior knowledge of the task, the robot's sensorimotor system, or its environment.

A small and general set of higher-level perceptual features appropriate to action in the agent's domain is found by unsupervised learning using a self-organizing feature map (Kohonen 1995). We use a variant SOM called the Growing Neural Gas (Fritzke 1995) that allows the number of units and the topology of the mesh to adapt to the properties of the domain. Using these learned features, the agent builds a set of high-level actions that carry it between *perceptually distinctive states* in the environment. SODA thus combines a *perceptual abstraction* of the agent's sensory input into useful perceptual features, and a *temporal abstraction* of the agent's motor output into extended, high-level actions, thereby reducing both the dimensionality and the diameter of the task.

Given high-dimensional, continuous-valued sensory input, and continuous motor output SODA works as follows.

1. Explore the environment with primitive ($\mathcal{A}^0$) actions, using a self-organizing feature map to learn a set of high-level perceptual features that define distinctive states in the environment. Figure 3(bot) shows examples of the learned perceptual features.

2. Learn a set of high-level ($\mathcal{A}^1$) actions in the form of control laws that carry the robot from one distinctive state to another. Each action consists of a trajectory-following control law that repeats a primitive action until a new perceptual feature becomes dominant, followed by a hill-climbing control law that maximizes the new dominant feature.

3. Use reinforcement learning in the abstracted space of high-level features and actions to learn a policy for the same high-diameter task, which now has much lower diameter with respect to the ($\mathcal{A}^1$) actions.

An experiment on a simulated robot navigation task (Figure 3) shows that the agent using SODA can learn to perform a task requiring 300 small-scale, local actions using as few as 9 autonomously-learned, temporally-extended, abstract actions. The learning time is substantially improved (Figure 4).

The methods discussed so far can learn the properties of the pixel-level sensorimotor system well enough to support autonomous learning of control laws and distinctive states suited to the environment the robot is embedded in. These distinctive states and the actions connecting them are the first steps toward a higher-level ontology for describing the robot's world. We now turn to two learning scenarios that build further on this higher-level ontology. First we look at the problem of *place recognition*: overcoming the variability of the pixel-level sensory image to recognize the current distinctive state directly and correctly from sensory input. And second, we take an important step toward learning the concept of *object*, a higher-level explanatory concept that make it possible to learn useful causal regularities about the world.

Bootstrap Learning for Place Recognition

Kuipers and Beeson (Kuipers & Beeson 2002) applied the bootstrap learning approach to the problem of learning to recognize places that may have originally been perceptually aliased. It is valuable for a robot to know its position and orientation with respect to a map of its environment. This allows it to plan actions and predict their results, using its map.

We define *place recognition* as identifying the current position and orientation in a large-scale space, a task sometimes called "global localization" (Thrun *et al.* 2001). However, not every location in the environment is a "place", deserving of independent recognition. Humans tend to remember places which are distinctive, for example by serving as decision points, better than intermediate points during travel (Lynch 1960).

We assume that the world and the agent's sensors are both very rich, so distinguishing information exists, but is hard to find. Real sensors are imperfect, so important but subtle image features may be buried in sensor noise. Two complementary problems stand in the way of reliable place recognition.

- *Perceptual aliasing*: different places may have similar or identical sensory images.
- *Image variability*: the same position and orientation may have different sensory images on different occasions, for example at different times of day.

These two problems trade off against each other. With relatively impoverished sensors (e.g., a sonar ring) many places have similar images, so the dominant problem is perceptual aliasing. With richer sensors such as vision or laser range-finders, discriminating features are more likely to be present in the image, but so are noise and dynamic changes, so the dominant problem for recognition becomes image variability. We want to use real sensors in real environments, avoiding assumptions that restrict us to certain types of sensors or make it difficult to scale up to large, non-simply-connected environments.

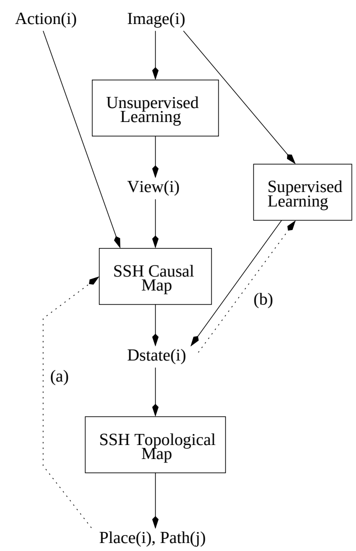


Figure 5: Bootstrap learning of place recognition. Solid arrows represent the major inference paths, while dotted arrows represent feedback.

When unique place recognition cannot be done using the current sensory image alone, active exploration will provide history information that can localize the robot and determine the correct place. However, when subtle features, adequate for discriminating between different places, are buried in the noise due to image variability, we want to recover those features.

We build on the abstraction of the continuous environment to a discrete set of *distinctive states* (dstates), provided by the *Spatial Semantic Hierarchy* (SSH) (Kuipers 2000). We assume that the agent has previously learned a set of features and control laws adequate to provide reliable transitions among a set of distinctive states in the environment (Pierce & Kuipers 1997).

The steps in our solution to the place recognition problem apply several different learning methods (Figure 5).

1. Restrict attention to recognizing *distinctive states* (dstates). Distinctive states are well-separated in the robot's state space.
2. Apply an unsupervised clustering algorithm to the sensory images obtained at the dstates in the environment. This reduces image variability by mapping different images of the same place into the same cluster, even at the cost of increasing perceptual aliasing by mapping images of different states into the same cluster. We define each cluster to be a *view*, in the sense of the SSH (Kuipers 2000).
3. Build the SSH causal and topological maps — sym-

bolic descriptions made up of dstates, views, places, and paths — by exploration and abduction from the observed sequence of views and actions (Kuipers 2000; Remolina & Kuipers 2004). This provides an unambiguous assignment of the correct dstate to each experienced image, which is feedback path (a) in Figure 5.

4. The correct causal/topological map labels each image with the correct dstate. Apply a supervised learning algorithm to learn a direct association from sensory image to dstate. The added information in supervised learning makes it possible to identify subtle discriminating features that were not distinguishable from noise by the unsupervised clustering algorithm. This is feedback path (b) in Figure 5.

We evaluated this method in experiments in two different real-world environments, one constructed to have a subtle distinguishing feature in an otherwise simple and symmetrical environment, and the other the main corridor in an office building. In both cases, unsupervised clustering produced significant amounts of perceptual aliasing, but with the help of the learned topological map, supervised learning was able to converge rapidly to 100% accurate place recognition.

This is a paradigm example of *bootstrap learning*. A weak learning method (k-means clustering) provides the prerequisites for an abductive method (topological map-building), which in turn provides the labels required by a stronger supervised learning method (nearest neighbor), which finally achieves high performance.

Bootstrap Learning of Object Representations

The blooming buzzing confusion of the pixel-level world is too variable to contain meaningful causal regularities useful for prediction and planning. Among the many important achievements in early childhood development is learning the higher-level concept of *object*, which along with higher-level actions is capable of supporting learning of causal regularities useful for understanding and manipulating the world (Spelke 1990).

In recent work toward this goal (Modayil & Kuipers 2004), we have shown how an agent can autonomously learn an ontology of *objects* to explain many aspects of its sensor input from an unknown dynamic world. For an agent to learn about an unknown world, it must learn to identify the objects in it, what their properties are, how they are classified, and how to recognize them.

The robot's sensorimotor system provides time-varying sensor inputs and motor outputs. From this, we assume that it can construct a description of the local environment in the "pixel-level" ontology of occupancy grid models.[2] The learning scenario described here takes place in "small-scale space", the space within the immediate sensory surround of the agent where it can reliably localize itself (Kuipers *et al.* 2004).

[2]The learning methods in (Pierce & Kuipers 1997) can learn the properties of sensors and effectors from experience. We assume that the occupancy grid representation and inference method can be learned in a similar way. We have a sketch of such a learning scenario, but it is outside the scope of this research on objects.

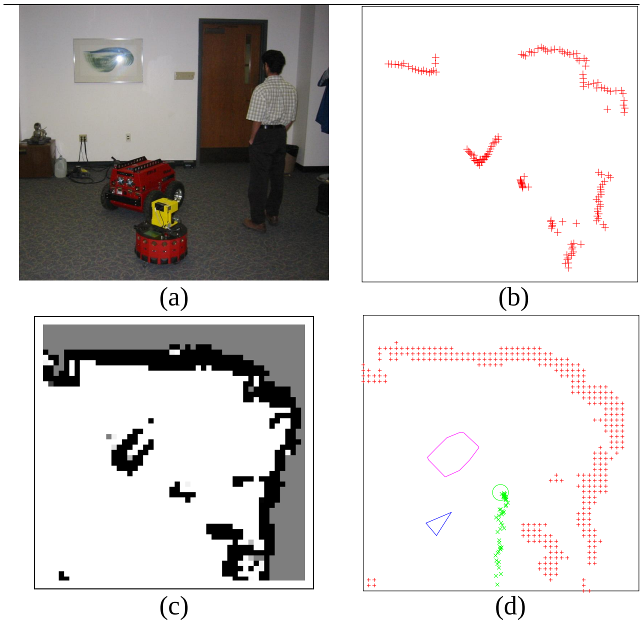


Figure 6: Multiple representations of a scene. The robot observer is the small round robot in the foreground. The larger ATRV-Jr is used as a non-moving object. **(a)**: A photograph of the scene. **(b)**: A range image of the scene at approximately the same time. **(c)**: An occupancy grid representation of the scene. **(d)**: An iconic representation of the scene. This is a symbolic description of the robot's environment enabled by the learned object ontology. The location of the observing robot is indicated by a small triangle (▷). A moving object (pedestrian) of amorphous shape is shown with its trajectory. A non-moving object (ATRV-Jr) has been classified, and is shown by the convex hull of its shape model. The permanently occupied cells in the occupancy grid represent the static environment.

The occupancy grid representation for local space does not include the concept of *object*. The occupancy grid representation assumes that the robot's environment can be divided into cells that are empty and those that are occupied. Evidence provided by range sensors is used to update the probability of occupancy of each cell. Simultaneous localization and mapping (SLAM) algorithms can efficiently construct an occupancy grid map and maintain accurate localization of a mobile robot within it using range sensor data (Moravec 1988; Thrun, Fox, & Burgard 2000).

In this bootstrap learning scenario, the learning agent acquires a working knowledge of *objects* from unsupervised sensorimotor experience. We begin by using the properties of occupancy grids to classify individual sensor readings as static or dynamic. A cell in the occupancy grid is considered *dynamic* if it changes from high-confidence occupied to high-confidence free, or vice-versa. A cell is considered *static* if it achieves and keeps a single high-confidence label, occupied or free. An individual sensor reading is labeled static or dynamic according to the label of the cell it falls in. Static readings are considered to be explained by the structure of the fixed environment.

The representation of objects is constructed from dynamic sensor readings in four steps: Individuation, Tracking, Image Description, and Categorization. Dynamic readings are clustered and the clusters are tracked over time to identify objects, separating them both from the background of the static environment and from the noise of unexplainable sensor readings. Once trackable clusters of sensor readings (i.e., objects) have been identified, we build shape models where they are stable and consistent properties of these objects. However, the representation can tolerate, represent, and track amorphous objects as well as those that have well-defined shape. The shape models are classified, so that instances of the same type of object can be categorized together.

In (Modayil & Kuipers 2004), we demonstrate this learning process using a mobile robot equipped with a laser range sensor, experiencing an indoor environment with significant amounts of dynamic change. The agent learned to individuate and track dynamic objects in the scene, acquired shape models where the shape was stable, and created a categorization of shape models. The scene could then be described in terms of the static environment (grounded to the static portions of the occupancy grid), and the dynamic objects (whose identities and trajectories could be described symbolically, grounded to the tracked objects in the scene). Figure 6 shows selected steps leading to this result.

Conclusions

To be autonomous, a robot must be able to learn its own ontology of higher-level concepts from its own pixel-level experience with the world, rather than obtaining it from an external programmer. We have described recent research that shows how the structure of unknown sensors and effectors can be learned (Pierce & Kuipers 1997); how high-level perceptual features and actions can be learned and used to define distinctive states (Provost, Kuipers, & Miikkulainen 2004); how high performance place recognition can be learned by bootstrapping unsupervised learning, map-building, and supervised learning (Kuipers & Beeson 2002); and how an ontology of objects can be learned from low-level experience with a dynamic world (Modayil & Kuipers 2004).

There are many other aspects of commonsense knowledge of the physical world still to be learned. We have already mentioned the need to learn the occupancy grid representation, or more generally, a local perceptual map representation of the immediate sensory surround (Kuipers *et al.* 2004). We are also extending the learned theory of objects with the actions that affect those objects, along with their preconditions and postconditions (Modayil & Kuipers 2004). Another important research direction will be learning to use vision as a sensory modality. Naturally, this kind of learning will straddle the evolutionary/developmental boundary.

Autonomous learning of the foundations of commonsense knowledge is one of the most critical research problems in artificial intelligence, and a rapidly-growing community of researchers is becoming aware of its importance.

References

Fritzke, B. 1995. A growing neural gas network learns topologies. In Tesauro, G.; Touretzky, D. S.; and Leen, T. K., eds., *Advances in Neural Information Processing Systems 7*, 625–632. MIT Press.

Kohonen, T. 1995. *Self-Organizing Maps*. Berlin; New York: Springer.

Kuipers, B., and Beeson, P. 2002. Bootstrap learning for place recognition. In *Proc. 18th National Conf. on Artificial Intelligence (AAAI-2002)*, 174–180. AAAI/MIT Press.

Kuipers, B.; Modayil, J.; Beeson, P.; MacMahon, M.; and Savelli, F. 2004. Local metrical and global topological maps in the hybrid spatial semantic hierarchy. In *IEEE Int. Conf. on Robotics & Automation (ICRA-04)*.

Kuipers, B. 2000. The Spatial Semantic Hierarchy. *Artificial Intelligence* 119:191–233.

Lakoff, G., and Johnson, M. 1980. *Metaphors We Live By*. Chicago: The University of Chicago Press.

Lynch, K. 1960. *The Image of the City*. Cambridge, MA: MIT Press.

McCarthy, J. 1968. Programs with common sense. In Minsky, M. L., ed., *Semantic Information Processing*. Cambridge, MA: MIT Press. 403–418.

Minsky, M. 1975. A framework for representing knowledge. In Winston, P. H., ed., *The Psychology of Computer Vision*. NY: McGraw-Hill.

Modayil, J., and Kuipers, B. 2004. Bootstrap learning for object discovery. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*.

Moravec, H. P. 1988. Sensor fusion in certainty grids for mobile robots. *AI Magazine* 61–74.

Piaget, J., and Inhelder, B. 1967. *The Child's Conception of Space*. New York: Norton. First published in French, 1948.

Pierce, D. M., and Kuipers, B. J. 1997. Map learning with uninterpreted sensors and effectors. *Artificial Intelligence* 92:169–227.

Provost, J.; Kuipers, B. J.; and Miikkulainen, R. 2004. Self-organizing distinctive-state abstraction learns perceptual features and actions. Submitted for publication.

Remolina, E., and Kuipers, B. 2004. Towards a general theory of topological maps. *Artificial Intelligence* 152:47–104.

Spelke, E. S. 1990. Principles of object perception. *Cognitive Science* 14:29–56.

Thrun, S.; Fox, D.; Burgard, W.; and Dellaert, F. 2001. Robust Monte Carlo localization for mobile robots. *Artificial Intelligence* 128:99–141.

Thrun, S.; Fox, D.; and Burgard, W. 2000. Monte Carlo localization with mixture proposal distribution. In *Proc. 17th National Conf. on Artificial Intelligence (AAAI-2000)*, 859–865. AAAI Press/The MIT Press.