Bootstrap Learning for Object Discovery

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

***Abstract*— We show how a robot can autonomously learn an ontology of *objects* to explain aspects of its sensor input from an unknown dynamic world. Unsupervised learning about objects is an important conceptual step in developmental learning, whereby the agent clusters observations across space and time to construct stable perceptual representations of objects. Our proposed unsupervised learning method uses the properties of allocentric occupancy grids to classify individual sensor readings as static or dynamic. Dynamic readings are clustered and the clusters are tracked over time to identify objects, separating them both from the background of the 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. In the end, the learned ontology makes it possible for the robot to describe a cluttered dynamic world with symbolic object descriptions along with a static environment model, both models grounded in sensory experience, and learned without external supervision.**

I. Introduction

Most work in robotics focuses on achieving competent behavior for a predetermined set of tasks. While this approach leads to reliable behaviors, the emphasis on performance and accuracy for engineered tasks leads to robots that lack any ability to learn about other aspects of the world. This paper examines how a robot can discover unknown objects, namely how a robot can perceive, track, model and recognize novel dynamic objects in the environment.

The goal of this work is to mimic the developmental learning process, where a learning agent must autonomously construct its own internal vocabulary to describe and interact with the world. To achieve this goal, we can not use complex algorithms that provide competence only for a limited set of objects and viewing circumstances. Instead, we rely on simple heuristics that provide an inclusive definition of objects so that the robot achieves some level of interaction with a broad range of objects. In future work we will examine how a robot can learn more sophisticated skills using statistics gathered from the perceived objects, but in this work we concentrate on demonstrating how a basic level of competency can be acquired.

For a robot 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 a "pixel-level" ontology of time-varying sensor inputs and motor outputs. Even after a substantial learning process [1] provides the organization on the sensors along with the ability to follow control laws and defines distinctive states to describe the large-scale structure of the environment, the robot's ontology still does not include *objects*. In this paper, starting from a lower-level ontology that includes range sensors, incremental motion, and an occupancy grid model of the local environment, we show how an ontology of objects can be learned without external supervision.

The occupancy grid representation for local space does not include the concept of "object." It assumes that the robot's environment is static, that it can be divided into locations that are empty and those that are occupied, and that the set of occupied locations has an arbitrary shape that can be detected by range sensors. A cell of an occupancy grid holds the probability that the corresponding region of the environment is occupied. 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 [2], [3], [4].

II. Learning about Objects

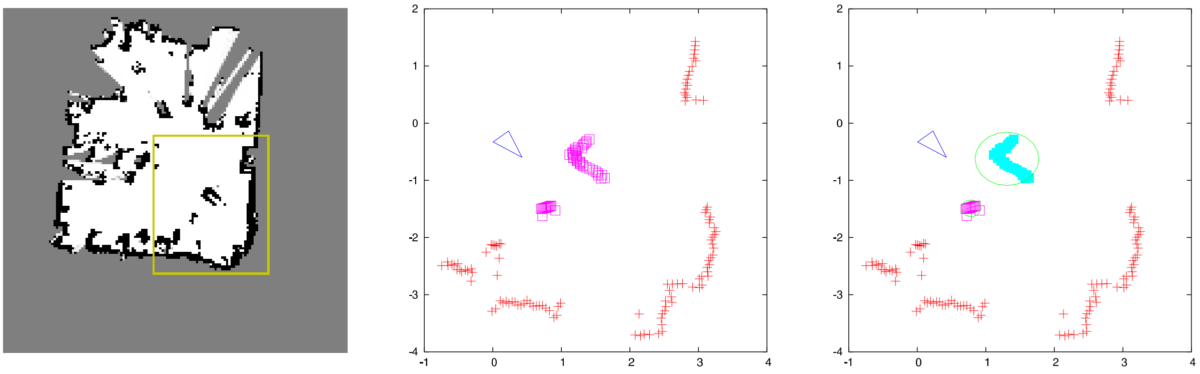
We claim that a robot can learn a working knowledge of *objects* from unsupervised sensorimotor experience by representing moveable objects in four steps: Individuation, Tracking, Image Description, and Categorization. We demonstrate this learning process using a mobile robot equipped with a laser range sensor, experiencing an indoor environment with significant amounts of dynamic change.

This is a kind of "bootstrap learning" [5] since we combine multiple learning methods, each learning the prerequisites for subsequent stages. For example, the object shapes learned here will be used in future work to recognize the same types of objects in more difficult contexts.

A major motivation for this work is to understand how complex cognitive structures can autonomously develop in a learning agent. We know that tremendous leaps in cognitive complexity occur through evolution and during infant development, using experience acquired in unconstrained environments.

Computational learning theory tells us that learning is exponentially hard in the dimensionality of the representation space [6]. Learning in a high dimensional representation space (such as an observation stream) should be

Proceedings of the 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems. pp. 742–747.
Sep 28 - Oct 2, 2004 Sendai, Japan.



(a) (b) (c)

Fig. 1. Object Individuation. (a) The occupancy grid representation of the environment generated online by a SLAM algorithm up to the current time t. The boxed region is shown in the following plots. (b) Sensor readings at time t classified as static $(+)$ or dynamic $(\Box)$ according to the occupancy grid cells they fall on. The robot $(\triangleright)$ is in the upper-left portion of the plot, so nearby dynamic objects occlude parts of the static environment. (c) Dynamic readings are clustered and hence individuated into objects. Each of the two clusters is assigned to a tracker (circles). [*All of these figures are clearer in the color PDF than in grayscale prints.*]

vastly harder than learning in a low dimensional (symbolic) representation. The premise of bootstrap learning is that an agent can apply a variety of high bias, but unsupervised learning algorithms to simple tasks (recognizing movable objects) to transform a high dimensional representation (an observation stream) into one with significantly lower dimension (a symbolic representation).

A. *Individuation*

The occupancy grid representation embodies a static world assumption. Sense data reflecting dynamic change in the environment are treated as noise. Fortunately, occupancy grid algorithms are quite robust to failures of the static world assumption. If changes in the environment are slow relative to repeated observation (12 Hz for the laser range-finder), changes in occupancy are quickly washed out by new observations, restoring the grid to a reasonably accurate description of the current state of the environment. We exploit this property and add a new attribute to the occupancy grid. A grid cell is labeled *transient* if it has ever been unoccupied (i.e., the probability of occupancy falls below a threshold), and *permanent* if it has never been unoccupied.[1]

The low-resolution occupancy grid cell labeling is used to classify individual high-resolution range sensor readings. Each individual range sensor reading is labeled as *static* or *dynamic*, depending on whether the endpoint of the reading falls in a cell labeled as permanent or transient, respectively. Permanent grid cells and static sensor readings represent the static background environment, and the learning algorithm restricts its attention to the dynamic range sensor readings. Note that a non-moving object such as a trash bin would be perceived with dynamic sensor readings if the robot had *ever* observed the space the readings are located in as unoccupied.

Next, the learning algorithm clusters the endpoints of the dynamic range sensor readings.[2] The coordinates of the endpoints x_i are represented in the fixed local frame of reference of the occupancy grid. Two endpoints are considered close if their distance is less than the threshold value δ_I:

$$close(x_i, x_j) \equiv \|x_i - x_j\| < \delta_I.$$

The individual clusters are the connected components of the $close$ relation: i.e., the equivalence classes of its transitive closure. Within a single observation frame at time t, these clusters $\{S_{i,t}\}$ are called *object snapshots*. They are the initial representation for individual objects. The process of individuation is shown in Figure 1.

B. *Tracking*

An object snapshot $S_{i,t}$ at time t has a spatial location and extent $<\!\mu_i, r_i\!>$: its center of mass μ_i and the distance r_i from its center of mass to its farthest reading. The dissimilarity between two snapshots S_i and S_j is

$$d_S(S_i, S_j) = \|\mu_i - \mu_j\| + |r_i - r_j|.$$

This function is robust to random noise and incorporates both the observed center and radius since the snapshots of a moving, dynamic object (such as a person) will vary in both dimensions. Where the successor to time t is t', we say that object snapshot S_t has *unique clear successor* $S'_{t'}$ if

$$d_S(S_t, S'_{t'}) < \delta_T$$

$$\forall S''_{t'} \neq S'_{t'} \quad d_S(S_t, S''_{t'}) > d_S(S_t, S'_{t'}) + \delta_R, \text{ and}$$

$$\forall S''_t \neq S_t \quad d_S(S''_t, S'_{t'}) > d_S(S_t, S'_{t'}) + \delta_R.$$

An *object tracker* is a function $T_k(t)$ whose value is an object snapshot $S_{i,t}$ at time t, such that for successive time-points t and t', $T_k(t')$ is the unique clear successor of $T_k(t)$. An object tracker T_k thus defines a collection of corresponding object snapshots extending from frame to

[1] To account for small localization errors, a transient cell may also require that all of its neighbors cells are unoccupied, which leaves permanent cells surrounded by a thin rim of unlabeled cells.

[2] Recall that the endpoints of range sensor readings, like the localization of the robot, are not limited to the resolution of the occupancy grid, but have real-valued coordinates, albeit with limited precision and accuracy.

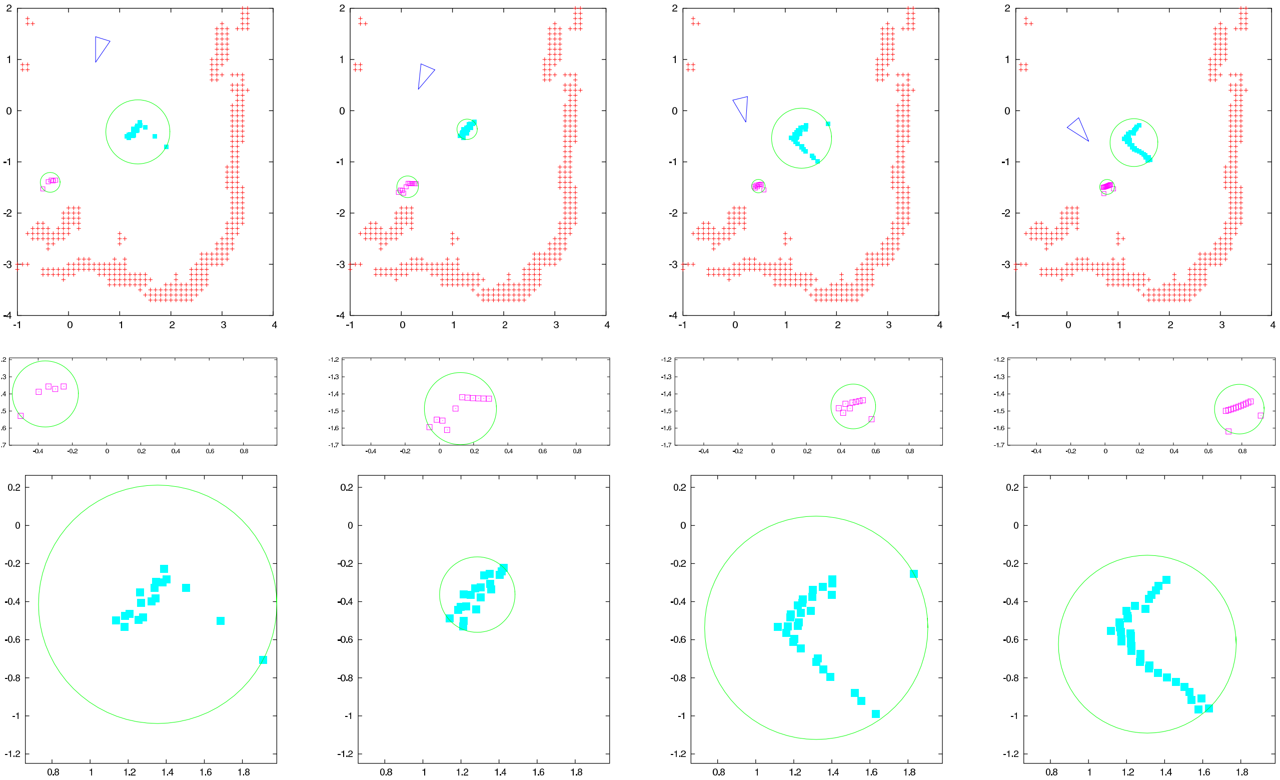


Fig. 2. Object Tracking. The shape of an object can vary greatly during tracking whether it has a rigid body or not. This figure shows a sequence of time steps prior to the scene in Figure 1. The actual trackers use data at much finer temporal granularity than the time-points (columns) shown. Note that the robot is moving while tracking. **Top**: The tracked dynamic objects, superimposed for reference on a low-intensity display of the permanent cells in the occupancy grid. **Middle**: A tracked pedestrian object, showing its irregular shape over time. **Bottom**: Tracked snapshots of a non-moving object (an ATRV-Jr).

frame in the observation stream, with at most one snapshot in each frame. The process of object tracking is depicted in Figure 2.

The domain of a particular object tracker ends at the time-points where the *unique clear successor* relation cannot be extended. "Object permanence", the ability of an object tracker to tolerate breaks in the sequence of frames, is clearly a learned ability in young children [7]. Our current implementation includes the ability to tolerate two missing frames in a sequence. Three missing frames terminates a tracker. New trackers are generated for large unexplained snapshots. Small snapshots without trackers are treated as noise and ignored.

Dynamic objects being tracked will converge and diverge, for example pedestrians in a crowded hallway. Object trackers will successfully track individuals over segments of their behavior, losing them when they get too close together and their readings are merged into a single snapshot. When they separate again, new trackers will be created to track the different individuals. More sophisticated methods for "object permanence" will be required to infer the identity of object trackers across such merges and splits. Following our bootstrap learning approach, we learn properties of objects during the periods of time when tracking is unambiguous and learning is easy. We expect those properties will make it possible to track objects under more difficult circumstances.

We define these trackable clusters of dynamic sensor readings to be objects. Each tracker represents a distinct symbolic identity which is assumed to be the cause of the readings associated with it. At this point, objects have only two properties: spatial location and temporal extent. These properties are sufficient for the trackers to guide the robot's actions to acquire additional information about the object. For example, control laws for following, circling and avoidance are easily specified using trackers to specify the desired goals. The next step will be to acquire properties of the object instances that are stable across changes in space and time. This makes it possible to categorize them into object classes.

C. Image Description

We have defined the *object snapshot* to be the set of sensor readings associated with an object at a particular time. The *shape model* for an object is a subset of the object snapshots collected over the time that the object is tracked.

The problem is how (and whether) the snapshots can be aggregated into a consistent, object-centered frame of reference. We consider it important to describe both objects with stable shapes that can be learned, and objects that are *amorphous* in the sense that they can be individuated and tracked, but their shape is beyond the capacity of the agent to describe and predict. For our robot learning agent, at its current level of sophistication, *pedestrians* are good examples of amorphous objects. At a later stage, the

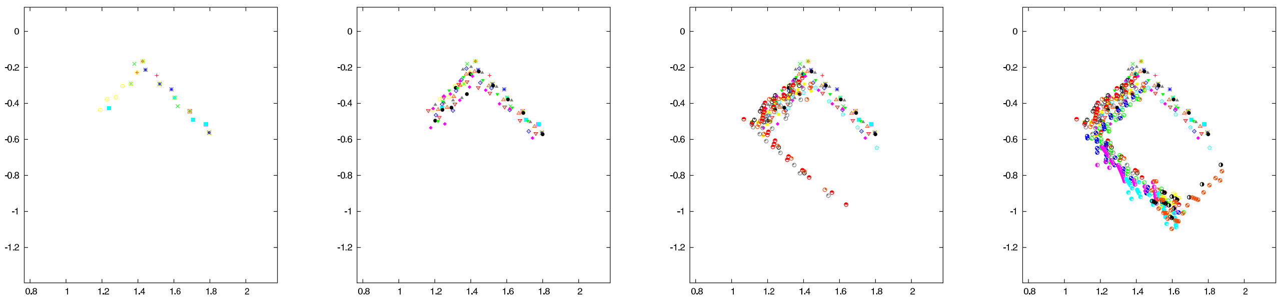


Fig. 3. Object Shape Model. This shows the incremental shape model creation for the ATRV-Jr observed in Figure 2. The range sensor endpoints in each snapshot are shown with different symbols. Selected snapshots combine to form a shape model.

learning agent may be able to model a pedestrian as two alternately-moving legs (observed as 2D blob shapes), but for now, object snapshots of pedestrians change too much to form stable shape models.

Consider a temporarily non-moving object such as an ATRV-Jr (a mobile robot). To be individuated and tracked as an object, it must be located at a position that was unoccupied at some time, so its sensor readings are considered dynamic. Since the object doesn't move in the environment, tracking is quite simple. However, as the robot moves around it, the object snapshot still changes slowly (Figure 2).

The agent creates a shape model by accumulating distinctive snapshots while the object appears to be non-moving (Figure 3). Both tasks, detecting the lack of object motion and determining distinctiveness, are accomplished by a non-symmetric dissimilarity function d_D that compares sets of points (snapshots).

$$d_D(S_{new}, S_{old}) = \frac{1}{|S_{new}|} \sum_{s \in S_{new}} \min(1, \frac{1}{\epsilon} \min_{t \in S_{old}} \|s - t\|)$$

When successive snapshots differ by a large amount, δ_M, the agent assumes the object has moved, and discards the current shape model. Otherwise, if the current snapshot is sufficiently distinct, δ_N, from the points currently in the shape model, the new snapshot is added to the shape model. Finally, snapshots in the shape model are discarded if they are incompatible with the full set of current sensor readings.

The shape model also records the directions from which the snapshots have been observed, and is considered *complete* when the full 360° surround has been sufficiently densely sampled.[3]

While the shape model is incomplete, it is considered "amorphous". When the shape model is complete, the agent creates a *standard shape image* for the object by placing the snapshots of the shape model into a canonical frame of reference. The snapshots are first rotated so that the primary axis of the readings is aligned with the y-axis. This is accomplished by rotating the shape model to minimize the entropy of the projection onto the x-axis. Next, the shape model is translated to minimize the distance of the farthest points from the origin. (See Figure 4.)

D. Categorization

Once an individual object has a standard shape image, the agent must categorize it. Note that the learning agent is responsible for building its own classes. Moreover, since the object observations come in incrementally, the agent must add new classes incrementally. The task of adding new classes incrementally is known as online clustering, and several algorithms exist [8]. For simplicity however, we solve this clustering task with a distance function.

We define the asymmetric dissimilarity function between two aligned shape images V and W by comparing their component snapshots

$$d'(V, W) = \frac{1}{|V|} \sum_{v \in V} \min_{w \in W} d_D(v, w).$$

We use this to define the symmetric distance measure

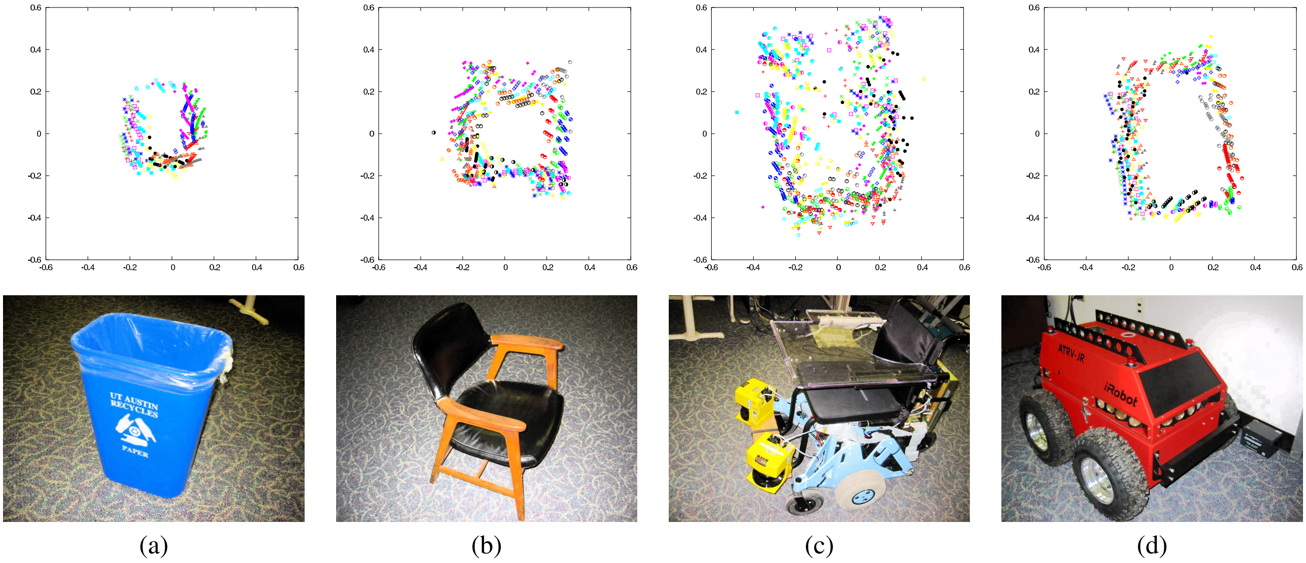
$$d_C(V, W) = \max(d'(V, W), d'(W, V)).$$

If the image of an instance is less than a threshold distance, δ_C, from multiple known types, then its classification is uncertain. If there is only one known type within δ_C, then it is classified as that type. If it is more than δ_C from any known type, then a new category is formed. For example, when the shape model in Figure 3 is converted into a standard shape image and compared to the known categories in Figure 4, it is recognized as an instance of the ATRV-Jr category. It is then displayed as a known type in Figure 5(d).

The robot does not learn a shape model by observing a continuously moving object, but it can learn a shape model if the object stops for a short period. Once an object has been classified, the tracker retains this classification and the corresponding shape model even when perception is difficult. Furthermore, the robot can obtain individual snapshots of a moving object, and we predict that those snapshots will be useful as evidence toward the classification of a moving object within an existing class hierarchy.

Even without a complete shape model, the robot can still generate a standard shape image for an object. For an incomplete image, the dissimilarity function is useful because it has the property that if $V \subset W$, then $d'(V, W) = 0$. This makes it suitable for comparing an incomplete model of an instance V with complete models that are already known. Also, this can be used to guide active perception by defining the observations that are most informative for classification.

[3] In the current implementation, this means at least one snapshot exists in each of six 60° pose buckets around the object.



(a) (b) (c) (d)

Fig. 4. Categorization entails both clustering and classification. Standard shape images and photographs for four learned object classes: (a) recycling bin, (b) chair, (c) robot wheelchair, and (d) an ATRV-Jr robot.

III. Experimental Results

The above system was implemented on an iRobot Magellan Pro robot equipped with a SICK PLS laser rangefinder. The parameters mentioned in the paper had the following values: $\delta_I = 0.5m$, $\delta_T = 1.0m$, $\delta_R = 0.01m$, $\delta_M = 0.5$, $\delta_N = 0.1$, and $\delta_C = 0.33$. The results of this experiment are not very sensitive to the selected parameter values and similar results arise when the parameters are varied by twenty percent.

The implementation was tested by running the robot in the lab. An occupancy grid representation of the environment (shown in Figure 1(a)) is generated online in the presence of object motions. The process of individuation is displayed in the subsequent two images, first showing the classification of laser scans as static or dynamic, and then clustering the dynamic readings to form snapshots. The snapshots are associated with trackers in Figure 2, providing temporal extent to the object representation. The ATRV-Jr robot is not moving during this time, so an image description is incrementally accumulated, as shown in Figure 3. When the description is sufficiently complete, the agent compares it to the objects learned earlier in the run, shown in Figure 4. The agent discovers that the image description best matches that of the ATRV-Jr robot.

The agent's world description is graphically represented in Figure 5 along with a photo of the same scene. The result is a discretization of a natural environment into several entities which are useful for later reasoning: a coarsely represented fixed environment (walls+furniture), a localized agent (the Magellan Pro robot), an amorphous moving object (a pedestrian), and a classified known object (the ATRV-Jr). This experiment demonstrates that object perception (individuation, tracking, description, and categorization) in cluttered environments is feasible even when using limited prior knowledge and simple representations. Moreover, since the agent can autonomously generate new categories online, its ability to succinctly and accurately describe nearby objects should improve with experience.

IV. Related Work

Other researchers have examined how object categories can be learned. Work on learning object classes in vision [9] demonstrates how a new category can be learned from a few examples. However, this work requires significant background knowledge of both generative models and priors on parameters. This background knowledge must be acquired from a more intensive learning process. In contrast to their work, we are concerned with how this complex background knowledge can be obtained autonomously by the robot. In future work, we intend to examine how the objects found by our work can provide background knowledge to bootstrap a system similar to theirs.

There is a large body of literature on individuation in both psychology and computer vision. Work in developmental psychology [7] suggests that infants learn Gestalt principles of perception. Work in perceptual psychology [10] demonstrates that the natural statistics of the environment can provide sufficient training data for acquiring grouping mechanisms. Individuation in vision has been achieved by a variety of criteria using the normalized cut algorithm [11].

Recent work on the Navlab project [12] has demonstrated the feasibility and value of tracking unknown objects in the environment. This work describes how a truck equipped with multiple range sensors is able to detect and track moving objects while driving down a road. The ability to track unknown moving objects is required for their goal of safe autonomous control at high speeds on urban streets. They are also able to recognize a couple of object classes. A significant difference from the work in this paper is their inability to generate new object types.

The construction of shape models of non-rigid objects has been explored in [13]. Using a variant of the iterative closest point algorithm, they are able to merge dense three-dimensional range scans into a single coherent shape

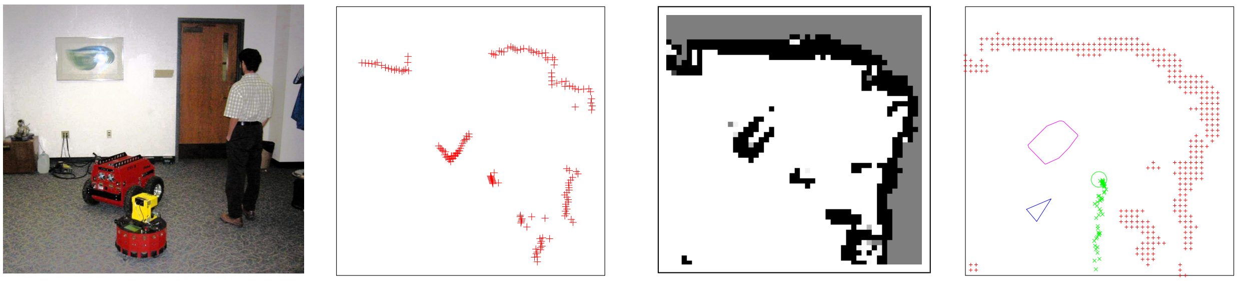


Fig. 5. Multiple representations of the scene in Figure 1. The robot observer is the small round robot in the foreground. The larger ATRV-Jr is used as a non-moving object. **(a)**: A photograph of the scene. **(b)**: A range scan representation of the scene. **(c)**: An occupancy grid representation 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 (as an instance of Figure 4(d)), and is shown by the convex hull of its shape model. The permanent cells in the occupancy grid are shown for reference, representing the static environment.

model even when the object undergoes small motions. This algorithm creates a qualitatively consistent model when an person moves their arms or head between successive scans. Because it relies on having significant amounts of data to align the scans, it is unclear that this method can be extended to handle non-rigid motion as observed by a two-dimensional range scanner.

In work by Biswas and colleagues [14], they create shape models from occupancy grids to generate new object types. They assume that the world is static during observation, which permits the use of a standard SLAM algorithm to capture the shape of the objects in a grid representation. The assumption that the entire environment stays static is fairly restrictive, since in many environments the objects of interest move regularly. Moreover, their algorithm uses an offline learning process. This makes the online incremental acquisition of new object types difficult.

V. Conclusions and Future Work

We have described and implemented a method for an agent to autonomously learn properties of novel dynamic objects in a natural environment without complex prior knowledge. This paper demonstrates how a learning agent can efficiently build an ontology of objects as part of a bootstrap learning process. Using this autonomously acquired ontology, a robot can categorize the dynamic objects it encounters in the world.

This work may be incrementally improved in multiple ways. Small errors in localization cause the shape models to become noisy, a problem that may be alleviated by better snapshot alignment. Also, the method is specified for a range sensor, so testing it with stereo vision is desirable.

An important part of bootstrap learning has not yet been explored here, namely utilizing acquired knowledge to construct informed priors to improve competence in harder tasks. This leads to several directions for future work: examining how class knowledge can aid in image description (by selecting discriminating observation angles), examining how image description can aid in tracking (by providing feedback on the plausible motion of the object), and using tracking to aid in individuation (by providing feedback for separating objects). Finally, we would like to examine how the learned object ontology can be used to speed up further learning tasks.

Acknowledgements

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.

References

[1] D. M. Pierce and B. J. Kuipers, "Map learning with uninterpreted sensors and effectors." *Artificial Intelligence*, vol. 92, pp. 169–227, 1997.

[2] H. P. Moravec, "Sensor fusion in certainty grids for mobile robots," *AI Magazine*, pp. 61–74, Summer 1988.

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

[4] A. Eliazar and R. Parr, "DP-SLAM: Fast, robust simultaneous localization and mapping without predetermined landmarks," in *Proc. 18th Int. Joint Conf. on Artificial Intelligence (IJCAI-03)*. Morgan Kaufmann, 2003, pp. 1135–1142.

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

[6] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. New York: Springer, 2001.

[7] E. S. Spelke, "Principles of object perception," *Cognitive Science*, vol. 14, pp. 29–56, 1990.

[8] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, Second ed. New York: John Wiley & Sons, Inc., 2001.

[9] F.-F. Li, R. Fergus, and P. Perona, "A Bayesian approach to unsupervised one-shot learning of object categories," in *Proc. Ninth IEEE ICCV*, 2003, pp. 1134–1141.

[10] W. S. Geisler and R. L. Diehl, "A Bayesian approach to the evolution of perceptual and cognitive systems," *Cognitive Science*, vol. 27, no. 3, pp. 379–402, May-June 2003.

[11] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888–905, 2000.

[12] C.-C. Wang, C. Thorpe, and S. Thrun, "Online simultaneous localization and mapping with detection and tracking of moving objects: theory and results from a ground vehicle in crowded urban areas," in *IEEE International Conference on Robotics and Automation*, 2003, pp. 842–849.

[13] D. Hähnel, S. Thrun, and W. Burgard, "An extension of the ICP algorithm for modeling nonrigid objects with mobile robots," in *Proc. 18th Int. Joint Conf. on Artificial Intelligence (IJCAI-03)*, 2003, pp. 915–920.

[14] R. Biswas, B. Limketkai, S. Sanner, and S. Thrun, "Towards object mapping in non-stationary environments with mobile robots," in *IROS*, 2002, pp. 1014–1019.