AAAI Spring Symposium Series 2013; *Designing Intelligent Robots: Reintegrating AI II*, Stanford, USA, March 25-27, 2013.

Integrating Visual Learning and Hierarchical Planning for Autonomy in Human-Robot Collaboration

Mohan Sridharan
Stochastic Estimation and Autonomous Robotics Lab
Department of Computer Science, Texas Tech University
Lubbock, TX 79409-3104
mohan.sridharan@ttu.edu

Abstract

Mobile robots deployed in real-world domains frequently find it difficult to process all sensor inputs, or to operate without human input and domain knowledge. At the same time, complex domains make it difficult to provide robots all relevant domain knowledge in advance, and humans are unlikely to have the time and expertise to provide elaborate and accurate feedback. This paper presents an integrated framework that creates novel opportunities for addressing these learning, adaptation and collaboration challenges associated with human-robot collaboration. The framework consists of hierarchical planning, bootstrap learning and online reinforcement learning algorithms that inform and guide each other. As a result, robots are able to make best use of sensor inputs, soliciting high-level feedback from non-expert humans when such feedback is necessary and available. All algorithms are evaluated in simulation and on wheeled robots in dynamic indoor domains.

1 Introduction

As we move towards deploying mobile robots in our homes, factories and other complex real-world domains where they have to collaborate with humans, we face formidable challenges such as autonomous operation, safety, engagement and interaction protocol design (Goodrich and Schultz 2007; Tapus, Mataric, and Scassellati 2007; Young et al. 2011). This paper focuses on autonomy in human-robot collaboration using sensor inputs and high-level feedback from non-expert humans. Real-world domains characterized by partial observability, non-determinism and unforeseen changes make it difficult for robots to operate without human feedback or domain knowledge. At the same time, robots cannot be equipped with all relevant domain knowledge in advance. Furthermore, humans are unlikely to have the time and expertise to interpret raw sensor inputs and provide elaborate and accurate feedback in complex domains. Many algorithms have been developed for robots to learn from sensor inputs or human training, and recent research has enabled *CoBots* and agents to use human feedback when needed or available (Rosenthal and Veloso 2012; Knox and Stone 2012). However, existing algorithms rely substantially on accurate domain knowledge or fail to model the unreliability of human input, limiting true autonomy to a small subset of (robot) capabilities in simulated or real-world domains. The objective thus is to answer the following questions:

- *How to best enable robots to adapt learning, sensing and processing to different scenarios and domains?*
- *How to best enable robots to seek high-level feedback from non-experts and merge it with information extracted from sensor inputs?*

While sophisticated algorithms have been developed for these learning, adaptation and collaboration challenges, the integration of these challenges for human-robot collaboration poses formidable open problems even as it presents novel opportunities to address the individual challenges. The framework described in this paper creates and exploits such opportunities by *jointly* addressing the associated challenges. The framework includes the following:

- **Hierarchical Planning:** algorithms integrate knowledge representation, non-monotonic logical inference and decision-theoretic planning, enabling robots to automatically adapt learning, sensing and information processing to the task at hand (Zhang, Sridharan, and Bao 2012).
- **Bootstrap Learning:** algorithms enable robots to use sensor inputs to autonomously learn models of domain objects and events, using the learned models to adapt to unforeseen changes (Li, Sridharan, and Meador 2013).
- **Augmented Reinforcement Learning:** algorithms enable robots and agents to merge unreliable human feedback with the information extracted from sensor inputs (Sridharan 2011).

These algorithms inform and guide each other, e.g., learning helps automate planning, while planning constrains learning to objects and events relevant to the task at hand. As a result, robots fully exploit relevant sensor inputs, soliciting high-level feedback from non-experts based on need and availability. This paper describes the framework, and illustrates the integration of learning and planning in the context of mobile robots using visual inputs and simplistic verbal cues to localize objects in indoor domains.

2 Related Work

The need for an integrated framework is established by discussing a representative set of related algorithms.

Sophisticated algorithms have been developed for segmentation, object recognition and learning of object models using visual cues and interactive feedback (Mikolajczyk and Schmid 2004; Parikh and Grauman 2011). These algorithms are used in conjunction with temporal cues, range data and textual tags for scene and activity understanding (Lei, Ren, and Fox 2012; Siddiquie and Gupta 2010). Similarly, algorithms for speech understanding have enabled robots to use verbal cues from humans in applications such as reconnaissance (Cantrell et al. 2010). In parallel, the constraints of classical planning algorithms have been relaxed to plan a sequence of actions in multiagent domains (Brenner and Nebel 2009). A non-monotonic logic programming paradigm such as answer set programming is ideal for common sense reasoning (Gelfond 2008; Chen et al. 2010), but it is not well-suited for probabilistic modeling of uncertainty, e.g., in sensing and navigation. On the other hand, probabilistic planning algorithms have helped deploy robots and agents in assistive scenarios and domains such as health care (Hoey et al. 2010; Rosenthal and Veloso 2012), but they make it difficult to represent and reason with common sense knowledge. Algorithms have also been developed for combining logical inference and probabilistic reasoning (Gobelbecker, Gretton, and Dearden 2011; Gogate and Domingos 2011; Richardson and Domingos 2006). However, revising domain knowledge using unreliable inputs, adaptation to unforeseen domain changes, and exploiting complementary properties of logical inference (e.g., default reasoning) and probabilistic planning remain open problems.

These learning and planning algorithms are used in conjunction with cognitive architectures (Hawes et al. 2010; CogX 2011) to bind information from sensor inputs, integrate cognitive models, and build spatial representations suitable for tasks such as reconnaissance (Cantrell et al. 2010). There is considerable focus on autonomous operation of socially assistive robots (Juan Fasola and Maja Mataric 2012), and on enabling robots to learn from demonstrations of domain experts (Cakmak and Thomaz 2012; Zang et al. 2010). Recent research is also enabling robots and agents to use human feedback when it is available or necessary (Rosenthal and Veloso 2012; Knox and Stone 2012). However, existing algorithms require accurate prior knowledge of specific task and domain, and do not fully account for the unreliability of human expertise and feedback, limiting true autonomy to a small subset of robot capabilities. Thus, adaptive sensor input processing and optimal use of unreliable feedback from non-expert human participants continue to be challenges to human-robot collaboration.

The integrated framework described in this paper seeks to exploit the dependencies between learning, adaptation and collaboration, creating novel opportunities to address the above-mentioned challenges. As a result, robots operate autonomously when possible, acquiring and using feedback from non-experts based on need and availability.

3 Integrated Framework

Figure 1 in an overview of the framework that integrates: (a) *Hierarchical planning* for acquiring and revising domain knowledge, combining non-monotonic logical inference and probabilistic planning to automatically adapt learning, sensing and processing to the task at hand; (b) *Bootstrap learning* for autonomously learning models of domain objects and events using local, global, temporal and contextual cues; and (c) *Reinforcement learning* for online merging of unreliable high-level feedback from non-expert humans with the information extracted from sensor inputs.

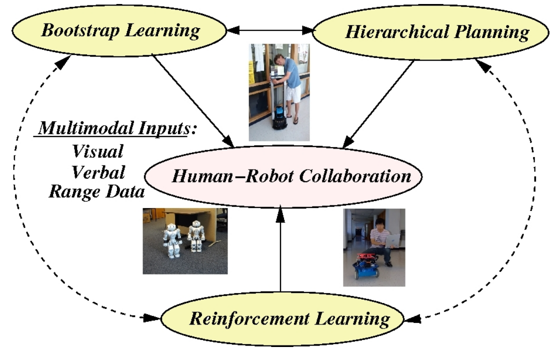


Figure 1: The framework exploits dependencies between learning, adaptation and collaboration to achieve autonomy in real-world human-robot collaboration.

These algorithms and their dependencies are described below, and the integration of learning and planning is illustrated in the context of mobile robots localizing (i.e., computing locations of) target objects in indoor domains.

3.1 Hierarchical Planning

Figure 2 shows the control loop of hierarchical planning. *Answer Set Programming* (ASP), a non-monotonic logic programming paradigm is used for knowledge representation and logical inference. An ASP program is a collection of statements describing domain objects and relations between them (Gelfond 2008). An *answer set* is a set of ground literals that represent beliefs of an agent associated with the program. Program consequences are statements that are true in all such belief sets. ASP readily supports *default reasoning* and includes concepts such as *default negation* and *epistemic disjunction*, e.g., unlike "`¬ a`", "`not a`" implies that "*a is not believed to be true*" and not that "*a is believed to be false*"; and "`p or ¬p`" is not a tautology. ASP is thus well suited for common sense reasoning and provides the appealing capability of non-monotonic reasoning—adding a new fact can reduce the set of (inferred) consequences. The *Knowledge Base* (KB) in ASP contains (common sense) rules and domain facts. Currently, rules are hand-coded and facts are learned incrementally from sensor inputs, human feedback and repositories—future work will investigate the incremental creation of rules. For any specific query or task, reasoning in the KB results in answer sets that represent current beliefs relevant to the query or task. An added advantage is that robots can acquire and store knowledge not directly relevant to the current task, which is typically a challenge in probabilistic planning schemes.

The uncertainty in sensing and navigation is modeled using hierarchical partially observable Markov decision processes (POMDPs). Beliefs are represented by probability

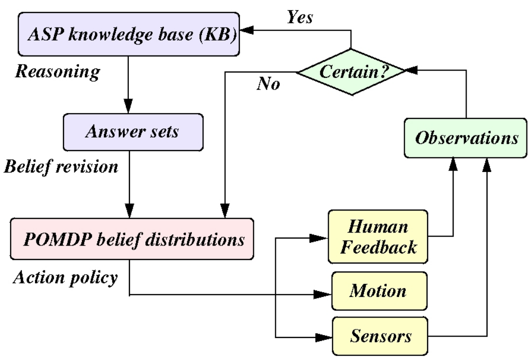


Figure 2: Hierarchical planning integrates knowledge representation, non-monotonic logical inference and probabilistic planning for human-robot collaboration.

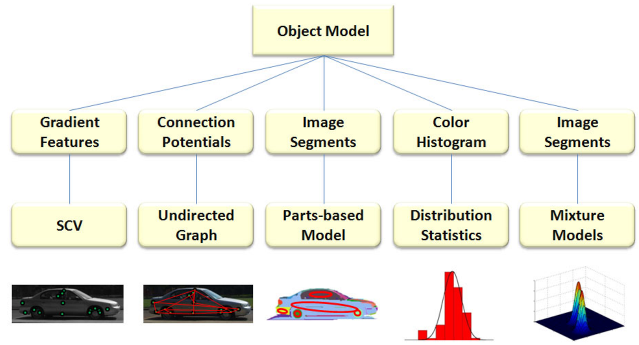


Figure 3: Learned object models use contextual and appearance-based cues to characterize objects.

distributions over the underlying states that are not observable. Our novel hierarchical decomposition includes convolutional policies, adaptive observation functions and learned (domain) models, enabling robots to reliably, efficiently and automatically create POMDP models, propagate beliefs, collaborate with teammates, and tailor sensing and processing to the task at hand (Zhang and Sridharan 2012).

For a specific task, answer sets are converted to (probabilistic) bias distributions using a psychophysics-inspired strategy that models object co-occurrence relationships. These distributions are merged with POMDP belief distributions and the learned POMDP policies are used to select actions at different levels of the hierarchy, i.e., to control movement, processing (e.g., analyzing images for learning and object recognition) and acquisition of human feedback. Robots obtain observations from sensors and human feedback; observations made with high certainty update the KB, while other observations update POMDP distributions. Robots solicit feedback when a human is available nearby and feedback is needed, e.g., if an object's location is known with considerable certainty, soliciting help to locate the object is not of much use. Robots visually identify humans and determine the need for feedback based on entropy of belief distributions. An instance of this hierarchical planning approach has enabled robots to collaborate with humans in complex indoor domains, planning actions to maximize information gain (Zhang, Sridharan, and Bao 2012).

3.2 Bootstrap Learning

Models of relevant domain objects and events are required to automate planning. Towards this objective, the framework includes bootstrap learning algorithms that enable robots to autonomously, incrementally and simultaneously: (a) learn probabilistic models of objects using visual cues; (b) use learned object models to learn models of visual features that characterize these objects; and (c) use feature models and object models to detect and adapt to unforeseen changes.

Figure 3 is an illustrative example of a model used to characterize domain objects. Since robots simultaneously learn the domain map and localize themselves in the map, objects that can move are considered to be *interesting*. Learning is triggered by motion cues based on the observation that characteristic features of an object have similar relative motion between consecutive images. Robots track local gradient features in short image sequences, identifying salient regions of interest (ROIs) corresponding to moving objects by clustering features with similar relative motion. Object models are then learned autonomously and incrementally using appearance-based and contextual visual features extracted from these ROIs. In Figure 3, object models consist of: relative spatial arrangement of gradient features; graphical models of neighborhoods of gradient features; parts-based models of image segments; color distributions; and probabilistic mixture models of local context, thus fully exploiting local, global, temporal and contextual cues. Robots use these object models in energy minimization algorithms and probabilistic generative models of information fusion, recognizing objects in novel scenes. The learned models are also used to incrementally learn models of the corresponding visual features (e.g., color and gradients). Robots track changes in these feature distributions to detect and adapt to changes in object configurations and environmental factors.

Robots using this bootstrap learning approach are able to reliably and efficiently learn and recognize objects in indoor (and outdoor) domains (Li, Sridharan, and Meador 2013). Since it is typically not practical to learn models of all objects using all image features, planning and human feedback are used to constrain learning to relevant objects and identify (most) informative features to characterize objects.

3.3 Reinforcement Learning

Robots with learning and planning capabilities can use human feedback to speed up learning, resolve ambiguities and revise domain knowledge. However, humans may not have the time and expertise to provide elaborate and accurate feedback in complex domains. The integrated framework includes augmented reinforcement learning (ARL) algorithms to merge unreliable feedback from non-experts with the information extracted from sensor inputs.

The ARL approach augments the traditional reinforcement learning formulation as shown in Figure 4. Integrating human feedback with environmental feedback obtained from sensors (H and R respectively in Figure 4) is a challenge because these feedback signals have different formats and are associated with changing levels of uncertainty over time. Unlike sensor inputs, human feedback can be a function of past (and even future) states and actions. Furthermore, robots may have to adapt to different humans and changes in human behavior over time. Unlike existing algorithms that assume human feedback to be accurate or model the uncertainty heuristically (Knox and Stone 2012), the ARL approach bootstraps off the two feedback signals to revise their relative contributions to the overall action policy. This strategy is motivated by the observation that many domains provide robots and humans shared access to performance measures such as task completion time and accuracy. The idea is to use these measures to incrementally and automatically revise the relative trust in the feedback mechanisms based on their relative ability to improve the robot's performance. The overall action policy in the ARL approach uses functions to merge R and H, revising function parameters continuously to optimize performance with the action policy. Furthermore, studies of human response times are used to learn a function that assigns the credit of human feedback to past states and actions.

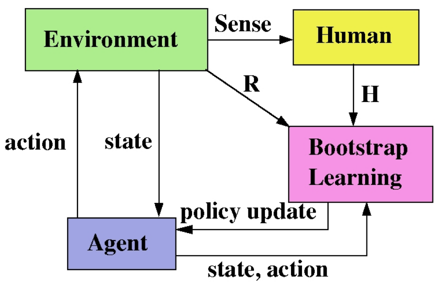


Figure 4: Online augmented reinforcement learning merges human feedback with environmental feedback.

Instances of the ARL approach have helped make best use of high-level human feedback in complex simulated domains with one or more agents (Sridharan 2011) and in adaptive interactions with mobile robots. However, an RL formulation typically requires knowledge of states and an estimate of transition and reward functions. The integrated framework will relax these limitations, defining: (a) states based on belief (and knowledge) states from planning; (b) rewards based on expected information gain; and (c) transition functions based on bootstrap-learned object models. This integration of ARL algorithm with learning and planning is work in progress but initial results are promising.

3.4 Integrating Learning and Planning

The integration of learning and planning is illustrated in the context of robots locating target objects in indoor domains. All algorithms were implemented using the Robot Operating System (ROS) (Quigley et al. 2009).

Figure 5 presents a subset of the architecture, with the visual bootstrap learning and hierarchical planning algorithms placed within the *vs_vision* node and *vs_planner* node respectively. The *vs_vision* node processes input images to provide the ID, relative distance and relative bearing of any detected object (along with a measure of certainty) to the *vs_planner* node. The *vs_planner* node, in turn, directs visual information processing for object recognition and learning of object models. Belief updates occur (in planning) when a robot arrives at a desired location and processes images of the scene, or processes images during navigation to a desired location. The planner node may also send coordinates of a relevant location to the movement control node *move_base* or direct the robot to solicit human input. The *move_base* node receives the current domain `map` from *map_server*, laser range information from *hokuyo_node*, and navigation goals (if any, e.g., from humans) through *navigation_goals*, in addition to pose and odometry information from *amcl* and (platform-specific) *erratic_base_driver* respectively. The *amcl* node performs localization to provide the pose estimate. The *move_base* node finds a path to the desired location and provides linear and angular velocity commands to the robot's driver. Additional nodes are created when required, e.g., for instances of other algorithms.

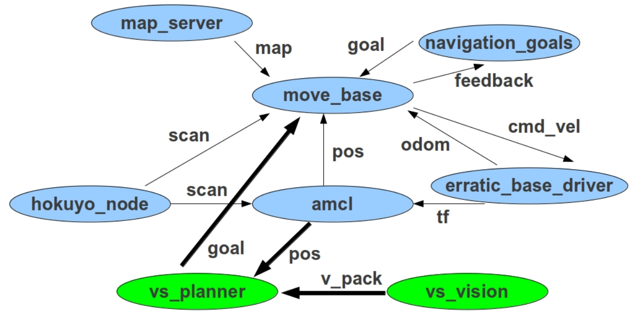


Figure 5: ROS used for integrating different components. Interaction between hierarchical planning, visual bootstrap learning and control modules is illustrated.

4 Experimental Results

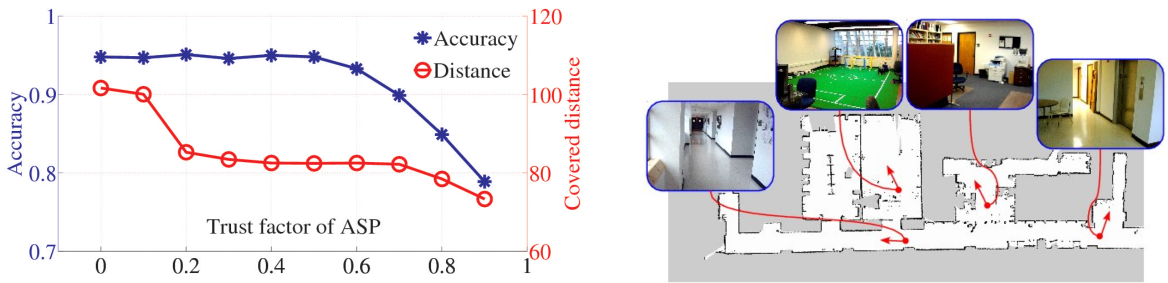
This section summarizes a subset of experimental trials conducted to evaluate the algorithms described above.

The bootstrap learning of object models was evaluated using $\approx$ 1400 images, including $\approx$ 700 images (captured by robots) of objects in motion, and images from the *Pascal VOC2006* benchmark dataset. The robot autonomously learned 30 different object models (i.e., subcategories) for objects in eight categories. To simulate challenging scenarios, each object model was learned using $\approx 3-5$ images. Test images consist of short sequences of objects in motion and images of objects in indoor and outdoor scenes. Multiple trials of learning and recognition were performed. Table 1 reports recognition accuracy averaged over subcategories (e.g., different "cars") in each category. Correct classification implies that test image objects are matched to the correct subcategory. Robots processed $3-5$ frames/second to learn models and recognize objects in novel scenes. The accuracy is high ($\approx 90\%$) despite the small number of images used for learning. Errors typically correspond to an insufficient number of test image features being matched with learned object models due to motion blur or a substantial difference in scale or viewpoint—incremental revision of object models eliminates some of these errors.

Hierarchical planning is evaluated on robots localizing objects in simulation and real-world domains. Domains with

	Box	Car	Human	Robot	Book	Airplane	Bus	Motorbike
Box	**0.958**	0	0.017	0.025	0	0	0	0
Car	0.010	**0.927**	0	0.021	0	0	0	0.042
Human	0.080	0.024	**0.820**	0.060	0.016	0	0	0
Robot	0.027	0	0.042	**0.899**	0.027	0	0	0.005
Book	0.016	0	0	0.042	**0.942**	0	0	0
Airplane	0.029	0.051	0	0.023	0.009	**0.888**	0	0
Bus	0	0.072	0	0	0	0	**0.856**	0.072
Motorbike	0	0.073	0	0.010	0.016	0	0.062	**0.839**

Table 1: Bootstrap learning of object models: accuracy $\in [0,1]$ averaged over subcategories in each category.



(a) ASP+POMDP.

(b) Learned domain map.

Figure 6: (a) Combining ASP and POMDPs provides high accuracy while significantly reducing the target localization time—trusting ASP beliefs too much has a detrimental effect on accuracy; (b) Learned domain map with offices, labs and corridors.

multiple rooms and objects were simulated and discretized into grids. Robots starting at random locations localized randomly selected targets; information about some other objects was provided as prior knowledge. Each point in Figure 6(a) is the average of 5000 simulated trials—x-axis represents level of relative trust in ASP-based beliefs. When ASP-based beliefs are not considered and only POMDP beliefs are used, accuracy is high but robots travel a significant distance (and spend considerable time) to localize targets. As ASP-based beliefs are included, distance traveled to localize targets decreases, and performance is stable over a range of trust factors. When ASP-based beliefs are trusted much more than POMDP beliefs, accuracy starts decreasing. Logical and probabilistic inference are thus equally important for reliable and efficient target localization. Results (not shown) also indicate that robots are able to solicit high-level human feedback based on need and availability—humans provide "yes/no" feedback or choose from multiple options regarding accessibility of rooms and likely object locations (Zhang, Sridharan, and Bao 2012).

The integration of learning and planning was evaluated on mobile robots in real-world domains, e.g., Figure 6(b) is the learned map of an entire floor in our department building with research labs, faculty offices, conference rooms and a kitchen. In 50 trials, robots autonomously revise domain map and object models, and (similar to simulated experiments) make best use of sensor inputs, domain knowledge and human feedback to reliably and efficiently localize targets. A video can be viewed online[1].

Figure 7 shows the performance of ARL in the $3vs2$ simulated keepaway (soccer) benchmark domain (Stone, Sutton, and Kuhlmann 2005), where three keepers maximize

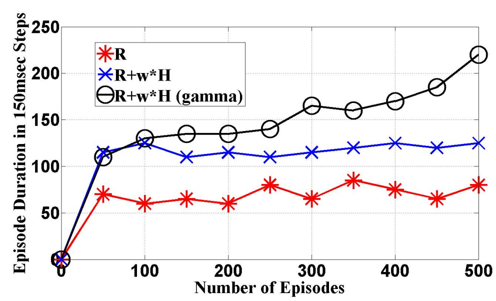


Figure 7: ARL makes best use of human and environmental feedback in 3vs2 keepaway—using learned gamma distribution for credit assignment increases episode length.

episode length by keeping the ball from two takers. Human and environmental feedbacks are merged by a linear function (i.e., weighted average) whose parameters are revised incrementally by bootstrap learning. Humans provide positive or negative reinforcement (e.g., "yes/no" feedback) no more than two times an episode, and one in every ten inputs is intentionally incorrect. ARL results in significantly longer episodes than the baseline $Sarsa(\lambda)$ RL algorithm. Using a gamma function for credit assignment further increases episode length. ARL also results in longer episodes than using just human feedback or other approaches for merging human and environmental feedbacks (Sridharan 2011).

5 Conclusions

This paper described an integrated framework that exploits dependencies between the learning, adaptation and collaboration challenges associated with human-robot collaboration, enabling robots to make best use of sensor inputs and high-level feedback from non-expert humans. Experimental results have been summarized for individual compo-

[1] `http://www.cs.ttu.edu/~smohan/Movies/Planning/aspPomdp.mp4`

nents and the integration of learning and planning. Future work will fully integrate ARL with learning and planning on robots. We are also integrating another component in this framework to learn multimodal associative models of objects and scenes, enabling robots to pose appropriate high-level verbal queries for human feedback. The ultimate goal is to enable widespread deployment of robots that can collaborate with humans in complex real-world domains.

Acknowledgments

Thanks to Shiqi Zhang, Xiang Li and Mamatha Aerolla for implementing the algorithms described in this paper. This work was supported in part by the ONR Science of Autonomy award N00014-09-1-0658.

References

Brenner, M., and Nebel, B. 2009. Continual Planning and Acting in Dynamic Multiagent Environments. *Journal of Autonomous Agents and Multiagent Systems* 19(3):297–331.

Cakmak, M., and Thomaz, A. 2012. Designing Robot Learners that Ask Good Questions. In *International Conference on Human-Robot Interaction (HRI)*.

Cantrell, R.; Scheutz, M.; Schermerhorn, P.; and Wu, X. 2010. Robust Spoken Instruction Understanding for HRI. In *International Conference on Human-Robot Interaction*.

Chen, X.; Ji, J.; Jiang, J.; Jin, G.; Wang, F.; and Xie, J. 2010. Developing High-Level Cognitive Functions for Service Robots. In *International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*.

CogX. 2011. Cognitive Systems that Self-Understand and Self-Extend. Retrieved Oct'12 from `http://cogx.eu/`.

Gelfond, M. 2008. Answer Sets. In *Handbook of Knowledge Representation*. Elsevier Science. 285–316.

Gobelbecker, M.; Gretton, C.; and Dearden, R. 2011. A Switching Planner for Combined Task and Observation Planning. In *AAAI Conference on Artificial Intelligence*.

Gogate, V., and Domingos, P. 2011. Probabilistic Theorem Proving. In *Uncertainty in Artificial Intelligence (UAI)*.

Goodrich, M. A., and Schultz, A. C. 2007. Human-Robot Interaction: A Survey. *Foundations and Trends in Human-Computer Interaction* 1(3):203–275.

Hawes, N.; Wyatt, J.; Sridharan, M.; Jacobsson, H.; Dearden, R.; Sloman, A.; and Kruijff, G.-J. 2010. Architecture and Representations. In *Cognitive Systems*, volume 8 of *Cognitive Systems Monographs*. Springer Berlin. 51–93.

Hoey, J.; Poupart, P.; Bertoldi, A.; Craig, T.; Boutilier, C.; and Mihailidis, A. 2010. Automated Handwashing Assistance For Persons With Dementia Using Video and a Partially Observable Markov Decision Process. *Computer Vision and Image Understanding* 114(5):503–519.

Juan Fasola and Maja Mataric. 2012. Using Socially Assistive Human-Robot Interaction to Motivate Physical Exercise for Older Adults. *Proceedings of IEEE* 100(8):2512–2526.

Knox, W., and Stone, P. 2012. Reinforcement Learning from Simultaneous Human and MDP Reward. In *International Conference on Autonomous Agents and Multiagent Systems*.

Lei, J.; Ren, X.; and Fox, D. 2012. Fine-Grained Kitchen Activity Recognition using RGB-D. In *ACM International Conference on Ubiquitous Computing (Short Paper)*.

Li, X.; Sridharan, M.; and Meador, C. 2013. Autonomous Learning of Visual Object Models on a Robot Using Context and Appearance Cues. In *International Conference on Autonomous Agents and Multiagent Systems*.

Mikolajczyk, K., and Schmid, C. 2004. Scale and Affine Invariant Interest Point Detectors. *International Journal of Computer Vision (IJCV)* 60(1):63–86.

Parikh, D., and Grauman, K. 2011. Interactively Building a Discriminative Vocabulary of Nameable Attributes. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 1681–1688.

Quigley, M.; Conley, K.; Gerkey, B. P.; Faust, J.; Foote, T.; Leibs, J.; Wheeler, R.; and Ng, A. Y. 2009. ROS: An Open-Source Robot Operating System. In *ICRA Workshop on Open Source Software*.

Richardson, M., and Domingos, P. 2006. Markov Logic Networks. *Machine Learning* 62(1-2):107–136.

Rosenthal, S., and Veloso, M. 2012. Mobile Robot Planning to Seek Help with Spatially Situated Tasks. In *National Conference on Artificial Intelligence*.

Siddiquie, B., and Gupta, A. 2010. Beyond Active Noun Tagging: Modeling Contextual Interactions for Multi-Class Active Learning. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*.

Sridharan, M. 2011. Augmented Reinforcement Learning for Interaction with Non-Expert Humans in Agent Domains. In *International Conference on Machine Learning Applications (ICMLA)*.

Stone, P.; Sutton, R.; and Kuhlmann, G. 2005. Reinforcement Learning for RoboCup Soccer Keepaway. *Adaptive Behavior* 13:165–188.

Tapus, A.; Mataric, M.; and Scassellati, B. 2007. The Grand Challenges in Socially Assistive Robotics. *Robotics and Automation Magazine, Special Issue on Grand Challenges in Robotics* 14(1):35–42.

Young, J.; Sung, J.; Voida, A.; Sharlin, E.; Igarashi, T.; Christensen, H.; and Grinter, R. 2011. Evaluating Human-Robot Interaction: Focusing on the Holistic Interaction Experience. *Social Robotics* 3(1):53–67.

Zang, P.; Irani, A.; Zhou, P.; Isbell, C.; and Thomaz, A. 2010. Using Training Regimens to Teach Expanding Function Approximators. In *International Conference on Autonomous Agents and Multiagent Systems*, 341–348.

Zhang, S., and Sridharan, M. 2012. Active Visual Sensing and Collaboration on Mobile Robots using Hierarchical POMDPs. In *International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*.

Zhang, S.; Sridharan, M.; and Bao, F. S. 2012. ASP+POMDP: Integrating Non-Monotonic Logic Programming and Probabilistic Planning on Robots. In *International Conference on Development and Learning (ICDL)*.