Integration of Robotic Perception, Action, and Memory

Li Yang Ku, Erik G. Learned-Miller, and Roderic A. Grupen
College of Information and Computer Sciences
University of Massachusetts Amherst, Amherst, MA
Email: {lku, elm, grupen}@cs.umass.edu

***Abstract*—In this paper, we address the interactions among robot perceptions, memories, and actions. We suggest that the ability to predict action consequences based on current perceptions and the memories of previous action consequences is essential for robots to behave intelligently in unstructured environments. Traditional approaches generally represent perception and action separately—as computer vision modules that recognize objects and as planners that execute actions based on labels and poses. We propose a more integrated approach in which a memory model integrates action and perception hierarchically and captures what the environment affords. This model can be learned efficiently through demonstrations. As more demonstrations are recorded and more interactions are observed, the robot becomes more capable of predicting the consequences of actions and thus better at planning sequences of actions to solve tasks in novel situations.**

I. Introduction

Humans and animals are remarkably adept at solving tasks in novel situations by generalizing past experiences to current observations in unstructured environments. This requires a joint understanding of perception, action, and memory. However, traditional approaches in robotics generally represent perception and action separately—as object models in computer vision and as action templates in robot controllers. Due to this separation, the robot can only interact with objects based on learned models when the object label is identified. Interacting based on object labels is not only vulnerable to recognition errors but also limits how past experiences can be generalized to novel situations.

In the book "On Intelligence", Jeff Hawkins asserts that *"Your brain receives patterns from the outside world, stores them as memories, and makes predictions by combining what it has seen before and what is happening now"*. The framework proposed here extends this concept and shows the capability of a memory model that integrates action and perception. This memory model represents how actions change observations and can be used to capture the affordances of the environment. With this integrated model, a robot would be capable of solving tasks by predicting perceptual action consequences based on memory and observation. This paper gives a broad overview of the proposed framework, and reviews our previous work that has investigated various components of it.

II. Related Work

The Memory-Prediction framework, a brain model that is consistent with neurological discoveries, is proposed by Hawkins in his book "On intelligence" [11]. This model emphasizes prediction from sequence memory based on the observation that humans recognize quotations and songs based on their sequences stored in memory. George and Hawkins further propose the Hierarchical Temporal Memory model that gives the Memory-Prediction framework mathematical foundations in Bayesian terms [8]. Lee and Mumford also suggest that based on findings on the early visual cortex activation, particle filtering and Bayesian-belief propagation algorithms might be used in cortical computations [25]. In this work, the concept of sequential memories is extended to recognizing objects. The relationship between a sequence of actions and a sequence of views are modeled not only to recognize objects, but also to provide robots with the capability to plan actions based on prediction. Next, we address the representation of objects in memory, and its relevance to actions.

In human psychophysics and neurophysiology, two models have been proposed to explain how objects are stored in human memory [42]. The object centered model represents each object by a small number of view-invariant primitives in an object centered reference frame [29]. Alternatively, viewer centered models represent each object as collections of viewpoint-specific local features. Since the development of these models, experiments in human psychophysics and neurophysiology have provided converging evidence for viewer centered models [5] [1]. Experiments on monkeys done by Logothetis et al. further confirm that a significant percentage of neurons in the inferior temporal cortex respond selectively to a subset of views of a known object [28]. The Aspect Transition Graph (ATG) used in our framework is a viewer centered memory model. In addition to distinctive views, an ATG summarizes how actions change viewpoints or the state of the object and thus, the observation. Besides visual sensors, extensions to tactile, auditory and other sensors also become possible with this representation. ATGs were first introduced in Sen's work [38] as an efficient way of storing knowledge of objects hierarchically. This work redefines the ATG as a directed multigraph composed of a set of aspect nodes connected by a set of action edges that capture the probabilistic transition between aspect nodes.

This ATG model in the proposed framework can memorize action outcomes and capture affordances of the environment. The term affordance has many definitions. We adopt the definition of aforadances as "the opportunities for action provided by a particular object or environment" [9]. Affordance can be used to explain how the "value" or "meaning" of things in the environment is perceived. The proposed framework is based on this interactionist view of perception and action that focuses on learning relationships between objects and actions specific to the robot. Some recent work in computer vision and robotics extended this concept of affordance and applied it to object classification and object manipulation [10] [14] [41] [43]. The proposed framework is based on affordances that are grounded in the robot's own actions and perceptions. Instead of defining object affordances from a human perspective, they are learned through direct interaction with objects from the robot's perspective.

Planning based on belief was introduced by Sondik and Smallwood for solving the optimal control problem characterized by the Partially Observable Markov Decision Processes (POMDPs) [40] [39]. The value iteration algorithm for solving POMDP was further improved by many authors such as [13] and [33] to solve larger problems. POMDP and the ATG memory model used in our framework have similar components. However, an ATG does not consider a reward function and is not used for finding optimal actions. The actions in an ATG are executed based on information in the states and do not belong to a fixed set of actions.

This proposed framework is tested on manipulation tasks that involve grasping. A lot of previous work has also been done on generating robotic grasp plans from visual information. In work done by Saxena et al., a single grasp point was identified using a probabilistic model on a set of visual features such as edges, textures, and colors [36]. Similar work uses contact, center of mass, and force closure properties based on point cloud and image information to calculate the probability of a hand configuration successfully grasping a novel object [37]. Platt et al. used online learning to associate different types of grasps with the object's height and width [35]. A shape template approach for grasping novel objects was also proposed by Herzog et al. [12]. A shape descriptor called a height map that captures local object geometry was used for matching part of a point cloud generated by a novel object to a known grasp template. Another work used a geometric approach for grasping novel objects based on point clouds [30]. An antipodal grasp was determined by finding cutting planes that satisfy geometric constraints. A similar approach based on local object geometry was also introduced [45]. In the work done by Lenz et al., a deep network trained on 1035 examples was used to determine a successful grasp based on RGB-D data [26]. Grasp positions were exhaustively searched and evaluated. Next, we discuss some of the relevant related work in neural networks.

Convolutional neural networks (CNNs) are a class of deep neural networks that contain more than one convolutional layer, and were introduced by Lecun and Bengio [24]. In the 2012 ImageNet Challenge, the CNN based approach proposed by Krizhevsky et al. generated results that surpassed other methods by a large margin [15]. CNN based approaches have since outperformed other approaches on most benchmarks in computer vision. Several authors have also applied CNNs to robotics. In the work done by Levine et al., visuomotor policies were learned using an end-to-end neural network that takes images and outputs joint torques [27]. A three layer CNN was used without any max pooling layer to maintain spatial information. In our framework, multiple convolution layers are also used; but unlike the previous work, relationships between layers are used to define a feature. Finn et al. used an autoencoder to learn spatial information of features of a neural network and demonstrate that the robot can learn tasks with reinforcement learning [7]. In [6], Finn and Levine further demonstrated that robots can learn to predict the consequences of pushing objects from different orientations and execute pushing actions to reach a given object pose based on a neural network structure with nine convolutional layers. In research done by Pinto and Gupta, a CNN was used to learn what features are graspable through 50 thousand trials collected using a Baxter robot [34]. The final CNN layer was used to select 1 out of 18 grasp orientations. The hierarchical CNN features used in our framework are based on CNNs trained on image classification, and hence require relatively little robot training data. This feature captures the hierarchical relationship between filters and can model local parts of a larger structure.

III. Framework

Figure 1 shows a modified conceptual diagram of the neocortex taken from the book "On Intelligence". Blocks with the same vertical positions represent neurons of the same cortex layer and arrows represent the direction of the information flow based on neuron connections. A neuron in a higher layer represents more abstract notions while a neuron in a lower layer represents simpler features. For example, visual neurons in a higher layer have larger receptive fields, represent object categories, and change more slowly over time. In this figure, memory regions that connect sensory neurons and motor neurons of the same layer are added to the original diagram. These memory regions associate neurons across modalities and can be used to infer bottom up signals that are missing. The connection loops within memory regions indicate predictions made based on observations, motor commands, and past memories. These memory regions have connections similar to the pyramidal neurons in the neocortex that have many connections within the same layer and an extended axon that sends signal to distant regions. However, these conjectured connections of the memory region are not based on neurological discoveries but on computational structures that have been shown to be practical in solving robotic tasks. The colored blocks and connections are implemented in the proposed framework and tested on robotic systems. In the following, we describe the memory model and the hierarchical structure in this diagram and show how they can be learned from demonstrations efficiently.

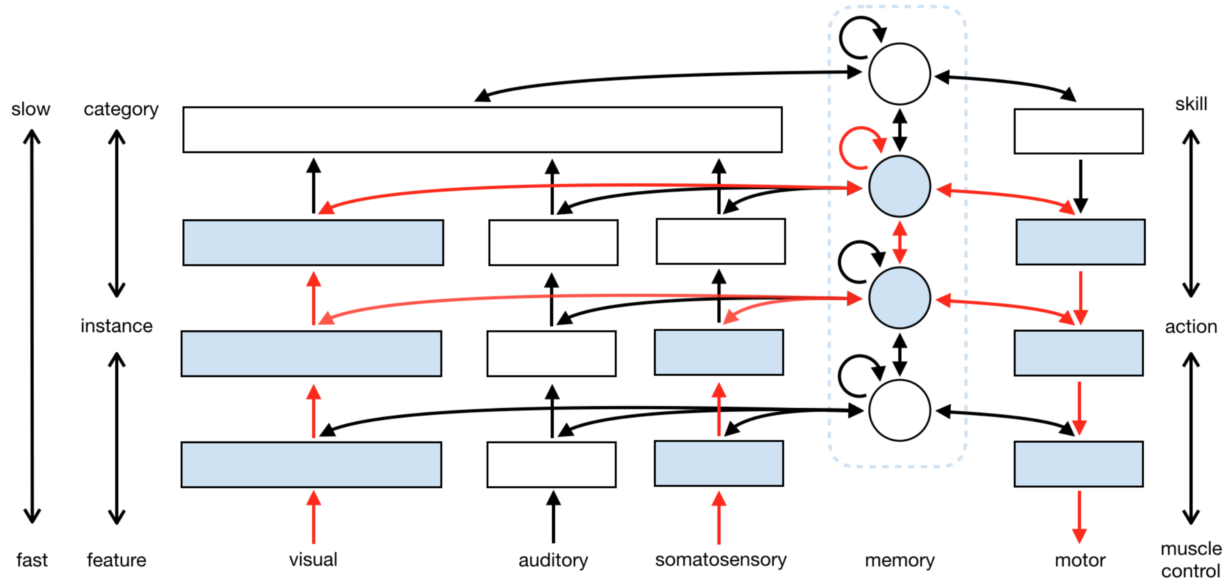

Fig. 1. A modified conceptual diagram of layers and connections in the neocortex where the highlighted memory regions are added to the original diagram introduced in the book "On Intelligence" [11]. The filled layers and red connections are implemented in the proposed framework.

A. Memory Model

In computer vision, there are two common types of object models used for identification. One represents objects in 2D and the other in 3D. However, neither of these incorporates information regarding how perceptions of objects change in response to actions. A robot that recognizes objects with traditional models knows nothing more than the label of the object. It is clear that humans have a different kind of object understanding—they can often predict the state and appearance of an object after an action.

Instead of an independent object recognition system, the proposed framework uses an integrated model called an *aspect transition graph* (ATG) that fuses information acquired from sensors and robot actions to achieve better recognition and understanding of the environment. An ATG is a memory model that memorizes past experiences about how actions change *aspects* (or observations stored in the model), and thus, maps observable states and actions to predicted future observable states.

An ATG is represented by a directed multigraph $G = (\mathcal{X}, \mathcal{U})$, composed of a set of aspect nodes $\mathcal{X}$ connected by a set of action edges $\mathcal{U}$ that capture the probabilistic transition between aspects. An action edge u is a triple (x_1, x_2, a) consisting of a source node x_1, a destination node x_2 and an action a that transitions between them. Note that there can be multiple action edges (associated with different actions) that transition between the same pair of nodes. Figure 2 shows a sample ATG model of a cube.

This memory model can be used to plan actions in partially observed environments. In previous work, we consider a simultaneous object modeling and recognition (SOMAR) task, where the robot has to model a given object while trying to recognize it [16]. An information theoretic planner that reduce uncertainty over objects by executing actions that maximally reduce the expected object entropy is proposed. The expected entropy is calculated based on the predicted action outcome stored in the ATG memory models. We showed that this approach outperforms a random action planner.

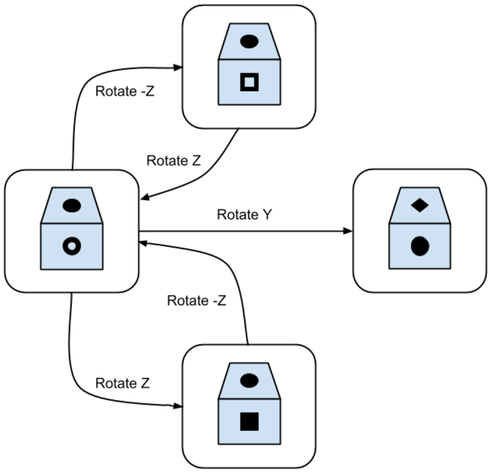

Fig. 2. Example of an incomplete aspect transition graph (ATG) of a cube object that has a pattern on each face. Each aspect is consists of observations of two faces of the cube. Each edge represents an action that transitions between observations.

The ATG model is also shown to be able to handle uncertainties in stochastic environments in previous work [19]. Through fine-grained transitions, we show that errors can be detected early by comparing observations with the predicted action outcomes. Transition probabilities are added to action edges in an ATG for actions that may result in random observations and errors can be handled accordingly. Surprising events that are not modeled in the memory are also handled by resetting the belief among aspects to the prior distribution; the robot would then re-examine the situation and identify possible solutions. We show that this approach results in more efficient actions and more robust results on a task that requires the robot to manipulate a box until it sees certain faces.

In [18], we introduced an ATG that considers a continuous observation space. Aspects are redefined as the set of observations within ϵ difference of a stored observation and the region of attraction is the set of observations that a closed-loop controller can converge to an aspect. Based on the funnel metaphor for closed-loop controllers introduced by Burridge [2], we introduce the slide metaphor for open-loop controllers that are used to represent action edges in an ATG model. A funnel may converge from a large set of robot states to a smaller subset, while a slide may end up in many different states due to noise. However, if a funnel-slide-funnel structure is constructed carefully such that the end of the slide is within the mouth of a funnel, we can guarantee a sequence of actions to succeed even when open-loop actions are included. Figure 3 shows the funnel-slide-funnel structure metaphor. This structure is tested on a tool grasping task where visual servoing is used to represent the funnel. We show that this structure reduces error significantly.

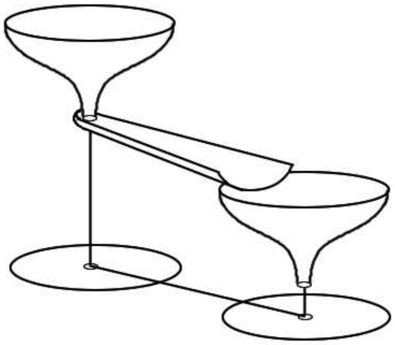

Fig. 3. Funnel-slide-funnel structure. The funnel metaphor is used to describe a closed-loop controller that converges to a subset of states, while the slide metaphor is used to describe an open-loop controller that causes state transitions.

Traditional grasping approaches such as the Willow Garage grasping pipeline [44] usually separates action planning from object recognition, where actions are executed based on object poses and labels generated from the vision module. In [17], we propose an alternative grasping approach where the observation is matched to the most similar aspect in the ATG memory model; actions are then executed based on action edges connected from this aspect. This approach does not require an explicit object pose of the object and allows the robot to act directly based on observation. We tested on a drill grasping task based on memorized grasping examples. Figure 4 shows that the robot grasps the drill differently based on the orientation.

B. Hierarchical Structure

Neural networks with hierarchical structures, such as Convolutional Neural Networks (CNNs), have outperformed other approaches on many benchmarks in computer vision. However applying them to robotics is nontrivial for two reasons. First, the final output of a CNN contains little location information, which is essential for manipulation. Second, collecting the quantity of robot data required to train a CNN is quite difficult.

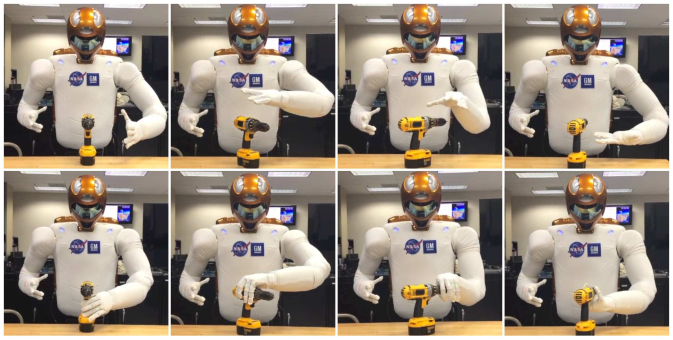

Fig. 4. Robonaut-2 grasping the drill posed at different orientations. Image pairs in the same row represents the intermediate and final states of one drill grasping trial.

The proposed framework tackles these challenges using the hierarchical CNN feature introduced in our previous work [21]. Hierarchical CNN features are extracted from a CNN trained on image classification therefore only require a small set of action examples. Instead of representing a feature with a single filter in a certain CNN layer, hierarchical CNN features use a tuple of filter indices to represent a feature. These features capture the hierarchical relationship between filters in different layers and can represent local parts of an object such as the right edge of the lower right corner of a box's top face. Hierarchical CNN features can be localized by back propagating filter responses along a single path to the input image and then mapped to a 3D point in the point cloud. This process traces backward recursively and yields a tree structure of hierarchical CNN features.

We consider a grasping task where the goal is to posture an anthropomorphic hand and arm for grasping based on visual information. A dataset consists of 120 grasping examples of six cylindrical and six cuboid objects is collected. Each example consists of the image, input point cloud, and joint configuration of the pregrasp pose. To map hierarchical CNN features to grasp pose, features that fire consistently are first identified among objects of the same class (cuboids or cylinders.) Features that have low offset variances to end effectors (index finger, thumb, and hand) among examples are then selected. By restricting the selected hierarchical CNN features to have the same high level filter, features will all be associated with the same object. Figure 5 show that without considering the hierarchical relationship, low level filters will fire on different objects in a cluttered scenario.

These selected hierarchical CNN features are then associated with a hierarchical controller that controls different kinematic subchains hierarchically. In this work, hierarchical CNN features in the fourth convolutional layer is associated with the arm controller and hierarchical CNN features in the third convolutional layer is associated with the hand controller.

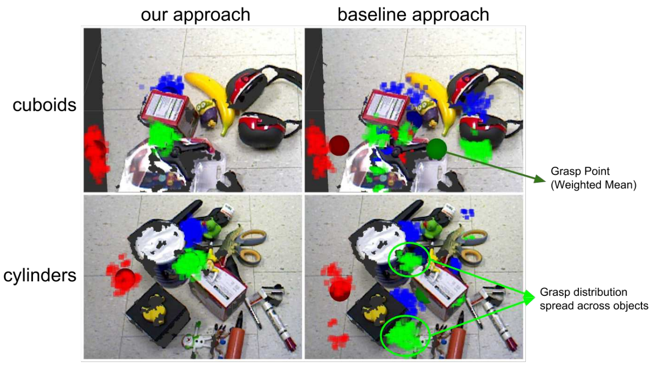

Fig. 5. Comparison in a cluttered scenario. The red, green, and blue dots represent proposed grasp points for the hand frame, thumb tip, and index finger tip of the left robot hand. Notice that the colored dots are scattered around in the baseline approach since the highest response filter in conv-3 or conv-4 layer are no longer restricted to the same high level structure.

The intuition behind these relations is that when moving the arm, a rough location of the object is sufficient and the detail object information is only needed when placing fingers. We evaluated this approach on 50 grasping trials on 10 novel objects and show significant improvement over a point cloud based approach.

This hierarchical CNN feature is further combined with proprioceptive feedback and force feedback to form a hierarchical aspect representation in [20]. This aspect representation is used to represent the stored observation in an ATG model and can be used to model the appearance, pose, and location of an object and the force feedback that the robot have perceived. This aspect representation is evaluated on the Washington RGB-D Objects dataset [23] on instance pose recognition and achieved state of the art result.

C. Learning from Demonstration

Learning from demonstration (LfD) is an attractive approach due to its similarity to how humans teach each other. However, most work on LfD has focused on learning the demonstrated motion [31], action constraints [32], and/or trajectory segments [4] [3] and has assumed that object poses can be identified correctly. This assumption may be true in industrial settings, but does not in general hold in unstructured environments.

In previous work [22], we present an integrated approach that treats identifying informative features as part of the learning process. This gives robots the capacity to manipulate objects without fiducial markers and to learn actions focused on salient parts of the object. Instead of defining actions as relative movements with respect to the object pose, our actions are based on spatial relationships between features. We classify demonstrations into three types: *a) robot-visual action* that specifies the target pose of a set of robot end effectors with respect to a set of 3-D visual feature locations, *b) robot-proprioceptive action* that specifies the target pose of a set of robot end effectors with respect to a set of current robot frames based on proprioceptive feedback, *c) visual-visual action* that specifies the goal position of a set of controllable visual features relative to another set of visual features on a different object. Based on the demonstration type provided by the operator, informative features that support actions can be identified automatically. Figure 6 shows the overall architecture. Through learning from demonstration, the ATG memory model, hierarchical aspect representation, and connections to the hierarchical controller can be learned together efficiently.

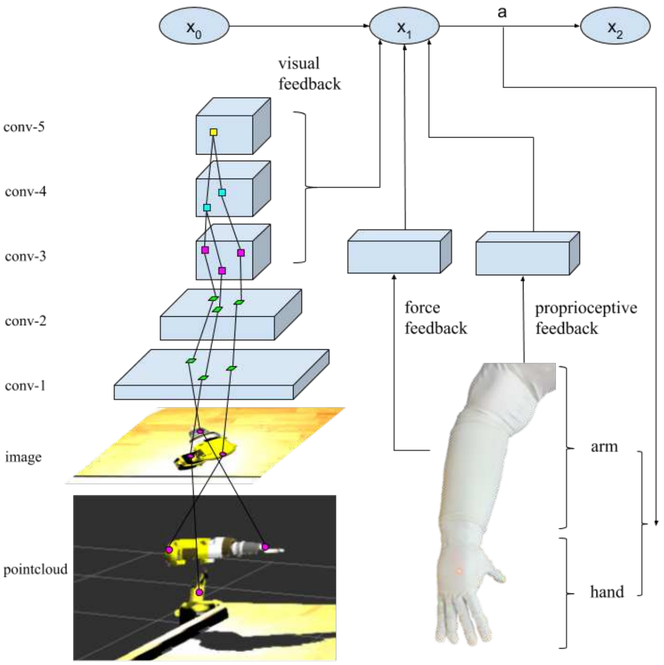

Fig. 6. The sensorimotor architecture driving transitions in the ATG framework. The aspect representation stored in an aspect node x is based on visual, force, and proprioceptive feedback. These information is used to parameterize action a for controlling the arm and hand motors.

However, the intent of the demonstrator may be ambiguous with a single demonstration. With multiple demonstrations, we show that ambiguities may be resolved by identifying consistent relationships between features. Figure 8 shows that through multiple demonstrations of mating the socket with the bolt, the robot is able to comprehend that the head of the ratchet should be aligned with the bolt autonomously.

This framework is demonstrated on a challenging bolt tightening task where the robot has to grasp the ratchet, tighten a bolt, and put the ratchet back into a tool holder with a small set of demonstrations. We show that the accuracy of mating the socket with the bolt can be increased with multiple examples. Figure 9 shows Robonaut-2 accomplishing this ratchet task. This learning from demonstrations approach is also tested on a drill grasping task in [20], where the goal is to grasp the drill on the handle with the robot's left hand. If the drill is out of reach, the robot has to plan a sequence of actions using both arms to extend its reachability based on grasping, rotating, and dragging actions learned from demonstrations. Figure 7 shows one of the trials that the robot executed both turning and dragging before grasping the drill.

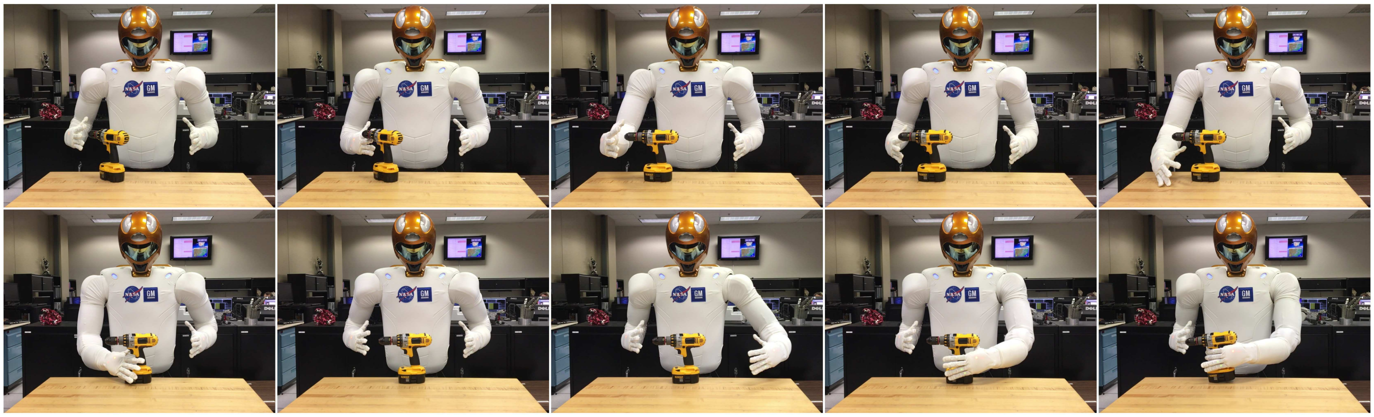

Fig. 7. Sequence of actions in one grasping test trial. The images are ordered from left to right then top to bottom. The initial pose of the drill is not graspable and located too far right for the left hand to reach. Therefore, the robot turns the drill then drags it to the center before grasping with its left hand.

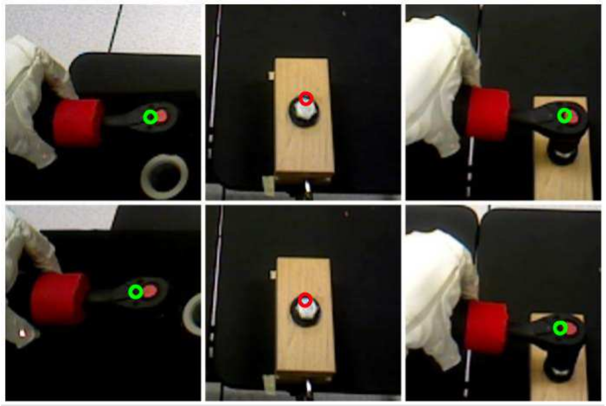

Fig. 8. Identifying informative features from multiple demonstrations. The two rows represent two demonstrations that place the socket of the ratchet on top of the bolt. The columns from left to right show the aspect nodes representing the tool, the target object, and the interaction. The green and red circles represent the most informative visual features selected for modeling the action.

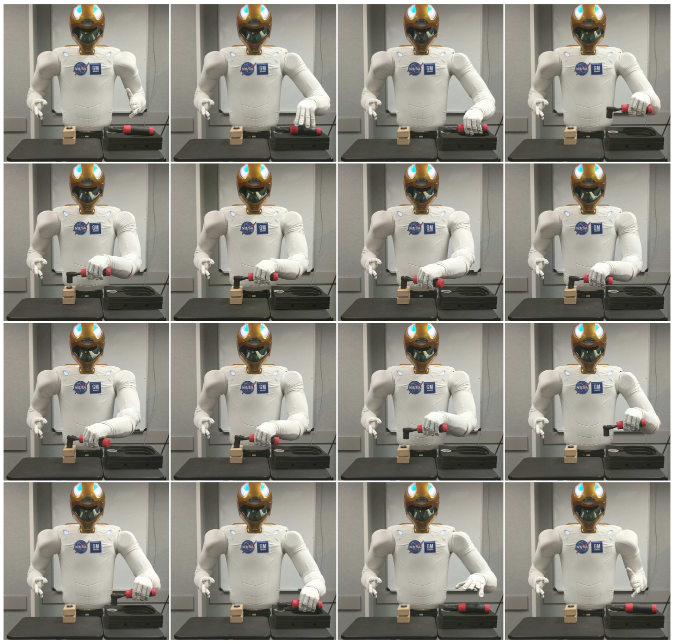

Fig. 9. The ratchet task sequence performed by Robonaut-2. The images from left to right, then top to bottom, show a sequence of actions where Robonaut-2 grasps the ratchet, tightens a bolt on a platform, and puts the ratchet back into a tool holder.

IV. Conclusions

The goal of this work is to present a framework that allows robots to solve tasks in an unstructured environment by predicting perceptual action consequences based on memory and observation. We have provided an overview of a series of works that explore parts of this framework.

A key component is the ATG memory model that memorizes action consequences through a directed multigraph composed of aspect nodes and action edges. By predicting action outcomes with this memory model, the robot can perform actions that help distinguish objects, detect errors early, reach goals reliably with a sequence of open-loop and closed-loop actions, and grasp objects without explicit pose estimation. We also presented a hierarchical structure that can be combined with this memory model based on hierarchical CNN features that are capable of representing local parts that belong to a high level structure. These features can be localized in 3D and are associated with a hierarchical controller to support grasping. We also explained how to combine ATG models with the proposed hierarchical structure by learning efficiently from demonstrations. We showed that through multiple demonstrations, informative visual features and consistent spatial relationships can be identified and used to model actions with higher accuracy.

Throughout this paper, we show that by predicting perceptual action consequences based on memory and perception, the proposed framework can accomplish a variety of challenging tasks under a unified framework. These results can be seen as support for the conjectured connections between sensory neurons, motor neurons, and memory regions in the proposed neocortex model of Figure 1. However, only a small part of this conceptual diagram is implemented. In future work, we would like to investigate the addition of more hierarchical relations in the memory model, consider cross modality inference, and learn models autonomously based on intrinsic motivation.

V. ACKNOWLEDGMENT

We are thankful to our colleagues Dirk Ruiken, Mitchell Hebert, Jay Wong, Michael Lanighan, Tiffany Liu, Takeshi Takahashi, and former members of the Laboratory for Perceptual Robotics for their contribution on the control basis framework code base. We are also grateful to Julia Badger, Philip Strawser, Jonathan Rogers, and the Robonaut Team for their support on Robonaut-2. This material is based upon work supported under Grant NASA-GCT-NNX12AR16A and a NASA Space Technology Research Fellowship. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Aeronautics and Space Administration.

REFERENCES

[1] Heinrich H Bülthoff and Shimon Edelman. Psychophysical support for a two-dimensional view interpolation theory of object recognition. *Proceedings of the National Academy of Sciences*, 89(1):60–64, 1992.

[2] Robert R Burridge, Alfred A Rizzi, and Daniel E Koditschek. Sequential composition of dynamically dexterous robot behaviors. *The International Journal of Robotics Research*, 18(6):534–555, 1999.

[3] Sylvain Calinon and Aude Billard. A probabilistic programming by demonstration framework handling constraints in joint space and task space. In *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*, pages 367–372. IEEE, 2008.

[4] Sylvain Calinon, Florent Guenter, and Aude Billard. On learning, representing, and generalizing a task in a humanoid robot. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 37(2):286–298, 2007.

[5] Shimon Edelman and Heinrich H Bülthoff. Orientation dependence in the recognition of familiar and novel views of three-dimensional objects. *Vision research*, 32 (12):2385–2400, 1992.

[6] Chelsea Finn and Sergey Levine. Deep visual foresight for planning robot motion. In *Robotics and Automation (ICRA), 2017 IEEE International Conference on*, pages 2786–2793. IEEE, 2017.

[7] Chelsea Finn, Xin Yu Tan, Yan Duan, Trevor Darrell, Sergey Levine, and Pieter Abbeel. Deep spatial autoencoders for visuomotor learning. *reconstruction*, 117 (117):240, 2015.

[8] Dileep George and Jeff Hawkins. Towards a mathematical theory of cortical micro-circuits. *PLoS Comput Biol*, 5(10):e1000532, 2009.

[9] James J Gibson. Perceiving, acting, and knowing: Toward an ecological psychology. *chap. The Theory of Affordance). Michigan: Lawrence Erlbaum Associates*, 1977.

[10] Helmut Grabner, Juergen Gall, and Luc Van Gool. What makes a chair a chair? In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 1529–1536. IEEE, 2011.

[11] Jeff Hawkins and Sandra Blakeslee. *On intelligence*. Macmillan, 2007.

[12] Alexander Herzog, Peter Pastor, Mrinal Kalakrishnan, Ludovic Righetti, Jeannette Bohg, Tamim Asfour, and Stefan Schaal. Learning of grasp selection based on shape-templates. *Autonomous Robots*, 36(1-2):51–65, 2014.

[13] Leslie Pack Kaelbling, Michael L Littman, and Anthony R Cassandra. Planning and acting in partially observable stochastic domains. *Artificial intelligence*, 101(1):99–134, 1998.

[14] Dov Katz, Arun Venkatraman, Moslem Kazemi, J Andrew Bagnell, and Anthony Stentz. Perceiving, learning, and exploiting object affordances for autonomous pile manipulation. *Autonomous Robots*, 37(4):369–382, 2014.

[15] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[16] Li Yang Ku, Shiraj Sen, Erik G Learned-Miller, and Roderic A Grupen. Action-based models for belief-space planning. *Workshop on Information-Based Grasp and Manipulation Planning, at Robotics: Science and Systems*, 2014.

[17] Li Yang Ku, Mitchell Hebert, Erik Learned-Miller, and Rod Grupen. Object manipulation based on memory and observation. In *First Workshop on Object Understanding for Interaction, at the International Conference on Computer Vision*, 2015.

[18] Li Yang Ku, Erik G Learned-Miller, and Roderic A Grupen. Modeling objects as aspect transition graphs to support manipulation. *International Symposium on Robotics Research*, 2015.

[19] Li Yang Ku, Dirk Ruiken, Erik Learned-Miller, and Roderic Grupen. Error detection and surprise in stochastic robot actions. In *Humanoid Robots (Humanoids), 2015 IEEE-RAS 15th International Conference on*, pages 1096–1101. IEEE, 2015.

[20] Li Yang Ku, Erik Learned-Miller, and Rod Grupen. An aspect representation for object manipulation based on convolutional neural networks. In *Robotics and Automation (ICRA), 2017 IEEE International Conference on*, pages 794–800. IEEE, 2017.

[21] Li Yang Ku, Erik G Learned-Miller, and Roderic A Grupen. Associating grasp configurations with hierarchical features in convolutional neural networks. In *Intelligent Robots and Systems (IROS), 2017 IEEE International Conference on*. IEEE, 2017.

[22] Li Yang Ku, Scott Jordan, Julia Badger, Erik G Learned-Miller, and Roderic A Grupen. Learning to use a ratchet by modeling spatial relations in demonstrations. *arXiv preprint*, 2018.

[23] Kevin Lai, Liefeng Bo, Xiaofeng Ren, and Dieter Fox. A large-scale hierarchical multi-view rgb-d object dataset. In *Robotics and Automation (ICRA), IEEE International Conference on*, pages 1817–1824, 2011.

[24] Yann LeCun and Yoshua Bengio. Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks*, 3361(10):1995, 1995.

[25] Tai Sing Lee and David Mumford. Hierarchical bayesian inference in the visual cortex. *JOSA A*, 20(7):1434–1448, 2003.

[26] Ian Lenz, Honglak Lee, and Ashutosh Saxena. Deep learning for detecting robotic grasps. *The International Journal of Robotics Research*, 34(4-5):705–724, 2015.

[27] Sergey Levine, Chelsea Finn, Trevor Darrell, and Pieter Abbeel. End-to-end training of deep visuomotor policies. *arXiv preprint arXiv:1504.00702*, 2015.

[28] Nikos K Logothetis, Jon Pauls, and Tomaso Poggio. Shape representation in the inferior temporal cortex of monkeys. *Current Biology*, 5(5):552–563, 1995.

[29] Thomas J Palmeri and Isabel Gauthier. Visual object understanding. *Nature Reviews Neuroscience*, 5(4):291–303, 2004.

[30] Andreas ten Pas and Robert Platt. Using geometry to detect grasps in 3d point clouds. *arXiv preprint arXiv:1501.03100*, 2015.

[31] Peter Pastor, Heiko Hoffmann, Tamim Asfour, and Stefan Schaal. Learning and generalization of motor skills by learning from demonstration. In *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, pages 763–768. IEEE, 2009.

[32] Claudia Pérez-D'Arpino and Julie A Shah. C-learn: Learning geometric constraints from demonstrations for multi-step manipulation in shared autonomy. In *IEEE International Conference on Robotics and Automation*, 2017.

[33] Joelle Pineau, Geoff Gordon, Sebastian Thrun, et al. Point-based value iteration: An anytime algorithm for pomdps. In *IJCAI*, volume 3, pages 1025–1032, 2003.

[34] Lerrel Pinto and Abhinav Gupta. Supersizing self-supervision: Learning to grasp from 50k tries and 700 robot hours. *arXiv preprint arXiv:1509.06825*, 2015.

[35] Robert Platt, Roderic A Grupen, and Andrew H Fagg. Re-using schematic grasping policies. In *Humanoid Robots, 2005 5th IEEE-RAS International Conference on*, pages 141–147. IEEE, 2005.

[36] Ashutosh Saxena, Justin Driemeyer, and Andrew Y Ng. Robotic grasping of novel objects using vision. *The International Journal of Robotics Research*, 27(2):157–173, 2008.

[37] Ashutosh Saxena, Lawson LS Wong, and Andrew Y Ng. Learning grasp strategies with partial shape information. In *AAAI*, volume 3, pages 1491–1494, 2008.

[38] Shiraj Sen. *Bridging the gap between autonomous skill learning and task-specific planning*. PhD thesis, University of Massachusetts Amherst, 2013.

[39] Richard D Smallwood and Edward J Sondik. The optimal control of partially observable markov processes over a finite horizon. *Operations Research*, 21(5):1071–1088, 1973.

[40] Edward Jay Sondik. The optimal control of partially observable markov processes. Technical report, DTIC Document, 1971.

[41] Alexander Stoytchev. Toward learning the binding affordances of objects: A behavior-grounded approach. In *Proceedings of AAAI Symposium on Developmental Robotics*, pages 17–22, 2005.

[42] Michael J Tarr and Heinrich H Bülthoff. Image-based object recognition in man, monkey and machine. *Cognition*, 67(1):1–20, 1998.

[43] Karthik Mahesh Varadarajan and Markus Vincze. Afrob: The affordance network ontology for robots. In *Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on*, pages 1343–1350. IEEE, 2012.

[44] Melonee Wise and Matei Ciocarlie. ICRA Manipulation Demo, 2010. URL http://wiki.ros.org/icra_manipulation_demo.

[45] Li Emma Zhang, Matei Ciocarlie, and Kaijen Hsiao. Grasp evaluation with graspable feature matching. In *RSS Workshop on Mobile Manipulation: Learning to Manipulate*, 2011.