

Active Learning using Mean Shift Optimization for Robot Grasping

Oliver Kroemer¹, Renaud Detry², Justus Piater^{2,1}, Jan Peters¹

¹Max Planck Institute for Biological Cybernetics, Spemannstr. 38, 72076 Tuebingen, Germany

²University of Liege, Grande Traverse 10, 4000 Liege, Belgium

email: {oliver.kroemer, jan.peters}@tuebingen.mpg.de,

{renaud.detry, justus.piater}@ULg.ac.be

Abstract—When children learn to grasp a new object, they often know several possible grasping points from observing a parent’s demonstration and subsequently learn better grasps by trial and error. From a machine learning point of view, this process is an active learning approach. In this paper, we present a new robot learning framework for reproducing this ability in robot grasping. For doing so, we chose a straightforward approach: first, the robot observes a few good grasps by demonstration and learns a value function for these grasps using Gaussian process regression. Subsequently, it chooses grasps which are optimal with respect to this value function using a mean-shift optimization approach, and tries them out on the real system. Upon every completed trial, the value function is updated, and in the following trials it is more likely to choose even better grasping points. This method exhibits fast learning due to the data-efficiency of the Gaussian process regression framework and the fact that the mean-shift method provides maxima of this cost function. Experiments were repeatedly carried out successfully on a real robot system. After less than sixty trials, our system has adapted its grasping policy to consistently exhibit successful grasps.

I. INTRODUCTION

Due to the detailed analysis of robot grasping in early works [1], [9], this area has a strong theoretical foundation. In the presence of good sensing, back-drivable actuation, and accurate models of both the grasped objects and the dynamics of the robot system, the appropriate grasp types, grasping forces and contact points characteristics can be automatically determined. Unfortunately, despite all progress, robot hands remain to be among the most difficult robot hardware to design and a robot hand that fulfills all the requirements above is currently not commercially available. Furthermore, an excessive number of object models would need to be readily available. As a result, researchers have begun embracing machine learning to support robotic grasping [10], [13], [16], [17].

The most straightforward and intuitive approach for teaching robots falls under the category of learning by imitation or programming by demonstration, a form of supervised robot learning. Learning by imitation involves showing the robot how to perform a given task by joy-sticking, kinesthetic teach-in, vision-based or SenSuit-based instruction. The robot subsequently attempts to repeat these motions to the best of its ability. Imitation learning for grasping suffers from several short-comings, i.e., as robot hands differ from human hands

both in kinematics and sensing, the demonstration itself is tedious and suffers strongly from the correspondence problem [6], and the learned behavior is limited to the teacher’s presentation and cannot adapt to new objects or situations.

Active learning approaches can help to address these issues [10], [16]. In active learning, the robot performs grasps on an object in order to refine its knowledge of how an object can be grasped. A teacher’s presentations can be used in order to initialize the process, thereby avoiding trying every possible grasp, and thus limit the active learning to the refinement of the teacher. The concept of combining learning from observations and trial & error resembles how a child would learn many manipulation tasks. The active learning approach suggested in this paper is aimed at robot grasping; however, it generalizes well and can be used in a variety of different settings. It makes use of two important components, i.e., (i) it uses a Gaussian process regression [15] in order to estimate the reward function of different grasps, and subsequently (ii) it employs mean-shift optimization [2] to find the best possible grasp candidates. As we only have few data points for an object and need fast generalization, this approach results in an efficient method.

The theory has been applied successfully in robot grasping using a real Mitsubishi PA-10 robot with a Barrett hand where it repeatedly converged on suitable grasp locations in less than sixty trials. At this point it had found two distinctly different yet stable ways of grasping the object while continuing to explore ways to improve these grasps. All experiments were carried out completely on the real robot system and no learning in simulation was required. This paper will proceed as follows: in the remainder of this section, we will first review existing work in Part I-A and show our assumed setting in Part I-B. The details of the active learning approach are given in Section II. The evaluations of the methods proposed in this paper are presented in Section III, followed by the conclusion in Section IV.

A. Related Work in Learning Robot Grasping

A large focus of previous works has been on learning grasp classification as success or failure based on visual cues. The choices of the features to test are often based on human intuitions and the earlier literature of grasp mechanics [1], [9]. Pioneering papers used mainly traditional neural networks, e.g., Moussa et al. [11] learned mappings from objects to

appropriate grasp types in a grasp hierarchy (although none of this work was evaluated on a real system), Pauli used radial basis-function networks to classify objects and recognize situations [12], Steffan et al. [19] have used tactile information to dynamically alter the grasp closing strategies based previously successful grasps which were represented by a self-organizing map (SOM) which approximated the grasp manifold [20].

To date, modern machine learning approaches such as support vector machines (SVM) and other kernel methods are frequently replacing neural networks as function approximators. Using these methods, several researchers have tackled the key problem of learning good points directly as a function of the shape features. Pelosso et al. [13] uses an analytical model to determine local features of a good grasp, and subsequently interpolates to untested situations in simulation. Recently, such approaches have been generalized such that both visual features and laser range finder data can be used for identifying the probability of success of a given grasp on a partially occluded object [18] and the algorithm is actually implemented on a real robot [17].

Another research direction, and one more aligned with the work presented here, is actively exploring an object to generate a full model of how to grasp the object. Fewer assumptions about the object being grasped, the kinematics of the hand, and the sensor system need to be made when grasps deemed successful have been experimentally proven to work on the robot. To obtain a complete model of grasp success probabilities, one should generally attempt new grasps at positions where the current model is lacking evidence, and therefore the model itself is uncertain. Salganicoff et al. [16] pioneered this direction using the confidence intervals of classification tree-based learning to determine which position to grasp in order to create more information rich models [16]. Morales et al. [10] used k-nearest-neighbors (KNN) for predicting the reliability of untested grasps, with the initial data being acquired by repeatedly attempting grasps on an actual robot system. They propose using the KNN to determine where the current model lacks information and test grasps in these regions to improve the model.

The algorithm applied here takes a different approach in that, rather than focusing on a complete model, the robot searches only for areas of the objects with good grasp affordances, which can then be used as a suitable grasping policy. This approach allows the robot to learn grasps faster and even directly on the real system, where the robot may also be trying to perform an object manipulation task for which it needs good grasps.

B. Visual Perception

All grasps are defined in the reference frame of the grasped object. Therefore, the first step towards active learning is determining the object position and orientation in the robot's reference frame. The robot uses a Videre STH-MDCS2-9cm stereo camera for the vision system. Tools such as laser range finders or sonar are not required for the pose estimation techniques applied here as it is purely based on standard stereo vision. The pose estimation software is the combination of the



(a) Vision



(b) ECV descriptors

Fig. 1: The top image shows the paddle on the stand taken from the left camera. The bottom image shows an image of the 3D reconstruction (model descriptors in green) generated from the stereo images of the same scene

Early-Cognitive-Vision (ECV) system [5] and a hierarchical Markov model [4]. The system is well suited for grasping experiments [3]. The vision system extracts edges and localizes them in five dimensions, i.e., three for position and two for orientation. The orientation along the edge can not be determined, and is not required [7]. The two colors along the edge are also stored as part of the feature. These features are usually called early cognitive vision (ECV) descriptors [14], and are used in generating models of objects as well as for observing scenes.

A model is generated by first extracting vision descriptors from stereo images of an object. If the images for initially creating the model were acquired from an unstructured scene, then the vision descriptors would be manually trimmed to only those generated by the object the system is focused on. These descriptors and their spatial relation to each other are encoded into a hierarchical Markov network of the object that has a tree structure with vision descriptors at the lowest level, and

the entire object at the top [4]. This probabilistic hierarchical Markov model is a full representation of the object. The object does not need to be represented by simpler objects such as cubes and spheres for this grasping task.

For the grasping task, the robot observes the scene and extracts its visual descriptors. The scene’s visual descriptors are not trimmed, but used directly as observations in the Markov model, allowing for the object’s six position and orientation to be inferred. Although individual vision features span a 5D sub-space, the combination of vision features, the object’s pose, and the final grasp exist in six dimensions of position (\mathbb{R}^3) and orientation (\mathbb{R}^3). As a probabilistic model, the system allows for the detection of multiple instances of the object in a given scene, and then selects the one with the highest likelihood [4]. Only one style of grasp generation will be addressed in the majority of this paper. Therefore, the term “grasp” will refer to a pose of the hand, regardless of whether it is successful in holding the object or not.

In the remainder of the paper, we will refer to the scene information as \mathbf{s} in $n\mathbb{R}^5$, and the grasp pose in the world reference frame as \mathbf{a} in \mathbb{R}^6 . The reward, $r(\mathbf{s}, \mathbf{a})$, of a grasp is determined by the amount the fingers need to adjust to the object during the lifting from the table. This method rewards grasps that are less dependent on the pose of the object on its stand.

It should be noted that the hardware used in this paper is more on the minimalist side for a grasping task, thus allowing the findings of this paper to be applied to a variety of different robots.

II. AN ACTIVE LEARNING ALGORITHM FOR ROBOT GRASPING

In this section, we propose an algorithm for actively learning good grasps for an object. The concept of active learning implies that the robot performs and tests grasps in the real world. It uses the gained knowledge to efficiently predict new and, possibly, better grasping poses for the next grasp. As discussed in Section I-B, we measure the performance using a reward $r(\mathbf{s}, \mathbf{a})$. The goal of the algorithm is to maximize the value function determined by the expected reward $J(\mathbf{s}, \mathbf{a}) = E\{r(\mathbf{s}, \mathbf{a})\}$ for an action in a particular state, which can be used for a grasping policy $\hat{\mathbf{a}} = \pi(\mathbf{s}) = \arg\max_{\mathbf{a}} J(\mathbf{s}, \mathbf{a})$.

In order to compute this policy efficiently, we need to solve two steps, i.e., approximate the value function J and find maxima \mathbf{a}^* efficiently. The algorithm estimates the value function of different grasp poses for an object using Gaussian processes regression. Mean-shift methods are applied to the value function in order to obtain a maximum. This maximum is subsequently tested by the real robot. As every robot trial is very costly, it is essential to make efficient use of them, and therefore the result is directly inserted into the reward function after each test, rather than applying a batch approach.

A. Value Function Approximation with Gaussian Processes

The value function, or expected rewards, of the grasps is estimated using Gaussian process regression [15], i.e., a Bayesian non-parametric regression approach that generalizes

well in the local vicinity of the training data. We will be using a Gaussian kernel with independent components given by

$$k(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-(\mathbf{x}_i - \mathbf{x}_j)^T \mathbf{H} (\mathbf{x}_i - \mathbf{x}_j)\right)$$

with $\mathbf{x}_i = [x_i, y_i, z_i, \theta_i, \phi_i, \psi_i]$ and $\mathbf{H} = \text{diag}(h_x, h_y, h_z, h_\theta, h_\phi, h_\psi)$, where \mathbf{x}_i and \mathbf{x}_j are the vectors containing the position and orientation variables of two grasps (i and j) in the space of the grasped object. Grasps in the object’s reference frame need to be determined from context \mathbf{s} and action candidate \mathbf{a} using kinematics, which we denote by $\mathbf{x} = f(\mathbf{s}, \mathbf{a})$ (note, that the inverse can also be determined). The hyper-parameters are contained within \mathbf{H} . If required, further variables can be added in here with ease. Given the initial data, suitable values for the hyper-parameters of the kernel function need to be constructed which can be achieved by maximizing the marginal likelihood. For more background on the optimization of hyper-parameters see [15]. The reward function is approximated by

$$J(\mathbf{s}, \mathbf{a}) = \sum_{i=1}^n \alpha_i k(\mathbf{x}_i, f(\mathbf{s}, \mathbf{a})), \quad (1)$$

using the n grasps observed to date where α_i denote the parameters or weights of this function. These weights are now determined using Gaussian process regression by $\alpha = (\mathbf{K} + \sigma_j^2 \mathbf{I})^{-1} \mathbf{r}$, where α denotes the vector of all α_i , \mathbf{K} is the so-called Gram matrix with entries $\mathbf{K}_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$, and the target vector \mathbf{r} contains the reward of all previous grasp candidates $r_i = r(\mathbf{s}_i, \mathbf{a}_i)$. The Gaussian process models the observations as having independent zero mean noise. The hyper-parameter σ_j^2 represents the variance of the observed values’ noise.

During the experiment, both successful and failed grasps are added to the Gaussian process regression in order to refine the reward function estimate. Upon the completion of each grasp, an additional data point is added and the weights need to be re-estimated. As only few trials should be performed, the sample size is limited, and thus the computational cost of this update is small in comparison and the storage requirements are negligible.

B. Determining Optimal Grasp Candidates using Mean-Shift Optimization

The main goal of this algorithm is to find good grasps without excessive exploration, and hence we need to determine promising grasp candidates based upon our current value function estimate. Finding optima in a multidimensional, non-convex function with many local optima is a generically hard task. However, due to the choice of value function representation in Section II-A, it can be tackled efficiently for grasping using the mean-shift optimization procedure. This procedure was originally designed to be used in kernel density estimation, but it transfers straightforwardly to reward functions represented by Gaussian processes [8]. It almost always converges onto a local maximum of the estimated value function. A full derivation and proof of convergence for the kernel density case can be found in [2]. The algorithm is iterative, so it needs to

be applied multiple times until convergence before the final grasp candidates can be determined.

It is straightforward to adapt the mean-shift algorithm to cost functions represented by Gaussian processes [8], where we initialize n particles in the standard way by $\mathbf{x}_k^0 = \mathbf{x}_k$ for $k \in \{1, 2, \dots, n\}$, and subsequently use the adapted update given by

$$\mathbf{x}_k^{t+1} = \frac{\sum_{i=1}^n \alpha_i (\mathbf{x}_i - \mathbf{x}_k) k(\mathbf{x}_k^t, \mathbf{x}_i)}{\sum_{i=1}^n \|\alpha_i\| k(\mathbf{x}_k^t, \mathbf{x}_i)} + \mathbf{x}_k^t \quad (2)$$

in order to update the particles. This update is iterated until convergence, and the final grasp is ready to be tested. Otherwise the new grasp coordinates are used to begin the next mean-shift iteration. While this loop is fairly expensive, mean-shift needs to be performed only once per trial, and is therefore not a large cost.

C. Exploring new Grasp Candidates

The mean-shift algorithm finds several local maxima. If only few trials have been performed, the reward function will not be sufficiently informative so that the best of these maxima can be determined. Instead, the system may stagnate by focusing too heavily on the first region with even mediocre rewards. Thus, it is essential to determine a strategy for exploring the different local maxima, each of which represents a grasp. A straightforward exploration strategy which usually works well in application is the Gibbs policy

$$\hat{\mathbf{a}} \sim p(\mathbf{a}_i^* | \mathbf{s}) = \frac{\exp(\tau J(\mathbf{s}, \mathbf{a}_i^*))}{\sum_{j=1}^q \exp(\tau J(\mathbf{s}, \mathbf{a}_j^*))},$$

where $\hat{\mathbf{a}}$ denotes the chosen maximum, \mathbf{a}_j^* denote the q maxima determined by the mean-shift algorithm and the temperature parameter τ can help trading off exploration and exploitation over time (although it was fixed in this setting). While other methods (e.g. ϵ -greedy) were considered, the Gibbs policy was used as it is a standard approach in reinforcement learning frameworks, and can also draw from the benefits of simulated annealing. Most of the details of the method for determining the next grasp to test have now been described, and a summary of the steps of the complete algorithm can be found in Algorithm 1.

III. ROBOT EVALUATIONS

In this section, we show how well the suggested approach performs in application on a real robot. First, in Section III-A, we show the robot setup used for our experiments and subsequently, in Section III-B, we show and discuss the results.

A. Experimental Setup

We use a hand eye system mounted on the wall consisting of (i) a Mitsubishi PA-10 medical light-weight robot arm with seven degrees of freedom, (ii) a three finger Barrett robot hand, (iii) a VEXTA pan tilt unit, and a (iv) Videre STH-MDCS2-9cm stereo camera system. The complete setup can be viewed in Figure 3. The experiment now proceeds as follows: first, the user selects a few possible locations for grasps as well as the grasp type, and then the active learning algorithm is started.

Algorithm 1 Active Learning of Grasps

Receive demonstrated grasps $\{x_1, x_2, \dots, x_n\}$

repeat

Observe scene and estimate the object's pose.

Estimate value function

$$\alpha = (\mathbf{K} + \sigma_f^2 \mathbf{I})^{-1} \mathbf{c},$$

$$J(\mathbf{s}, \mathbf{a}) = \sum_{i=1}^n \alpha_i k(\mathbf{x}_i, f(\mathbf{s}, \mathbf{a})),$$

Initialize n particles by $\mathbf{x}_k^0 = \mathbf{x}_k$.

repeat

Update all particles by

$$\mathbf{x}_k^{t+1} = \frac{\sum_{i=1}^n \alpha_i (\mathbf{x}_i - \mathbf{x}_k) k(\mathbf{x}_k^t, \mathbf{x}_i)}{\sum_{i=1}^n \|\alpha_i\| k(\mathbf{x}_k^t, \mathbf{x}_i)} + \mathbf{x}_k^t$$

until convergence $\forall k. |\mathbf{x}_k^{t+1} - \mathbf{x}_k^t| < \epsilon$

if all particles have the same value

Choose the corresponding grasp.

else there is more than one mode

Draw one of the particles according to

$$\hat{\mathbf{a}} \sim p(\mathbf{a}_i^* | \mathbf{s}) = \frac{\exp(\tau J(\mathbf{s}, \mathbf{a}_i^*))}{\sum_{j=1}^q \exp(\tau J(\mathbf{s}, \mathbf{a}_j^*))}$$

end

Execute the grasp on the robot system.

Observe the reward r_{n+1} .

Insert data point $(\mathbf{x}_{n+1}, r_{n+1})$ into the GP.

until user-intervention

As an object to be grasped, we chose a table tennis racket and the user was allowed to select a few initial grasping locations, orientations and grasp types (mainly precision pinches) by visual inspection and joy-sticking the robot, only limited by the fact that they need to be reachable by the robot. The demonstrator chose mainly grasps at three locations, i.e., one group of grasps was focused on the handle, and two further ones were placed on the flat section. The grasps were demonstrated through joy-sticking.

Initial imitation attempts tend to display low success rate due to the method's inherent problems: the hand was joy-sticked remotely from the operators desk (as a result, the scene understanding of the operator was limited), lack of experience from the demonstrator for this type of hand, and that the Barrett hand is not equipped with tactile sensors nor with proper joint torque sensors. As a result, the grasps closure program needs to rely purely upon an open loop program that upon completion determines whether the hand has grasped the object by attempting to lift the object. If it can be steadily lifted, the grasp is labeled a success. Thus, the human presentations are imperfect and it is often hard for the human operator to achieve a high success rate.

For autonomous active learning, the robot must itself determine where to grasp and improve on a trial by trial basis. A trial consists of the following steps: first, the robot observes the scene and estimates the object's pose, and the mean-shift optimization is employed to generate a new grasp based on all prior knowledge on possible grasping points. The robot then moves its hand towards the grasp position along the desired grasping approach direction. Once the appropriate position is reached, the hand starts the grasping program, grasps the object and tests the grasp before returning to the

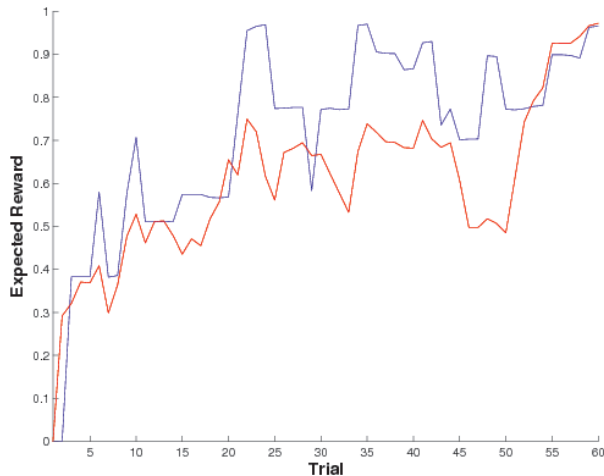


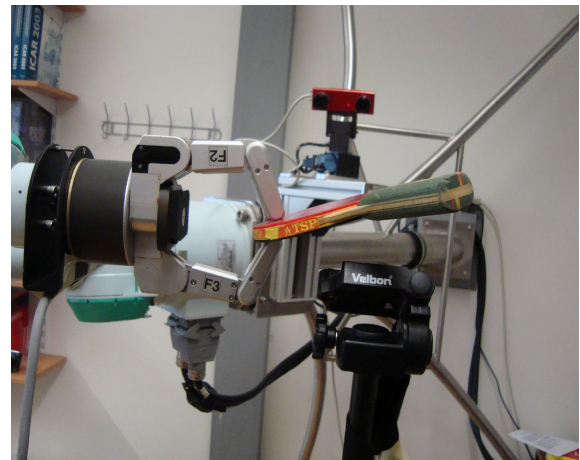
Fig. 2: The **red** plot shows the expected reward, which is increasing with the number of trials. The **blue** plot is an example trial, deriving the expected reward with a uniform five-point filter. Over all experiments, the last five trials had mean rewards of 0.9418 with a standard deviation of only 0.0981

home position. All experiments were carried out on the real system and no simulation was required.

B. Results and Discussion

The experiment was repeated four times, using different sets of initial grasps, and were run for 60 trials each. The system presented a gradually increasing overall expected reward shown in Figure 2. At the end of the experiments, the robot had successfully achieved a state of high expected rewards with low uncertainty. The small drops in the expected reward and sometimes moderate improvements are artifacts resulting from several sources, i.e., (i) that the applied grasp was not a sufficiently robust grasp, and (ii) that as long as the robot is still exploring new grasps, it will inevitably try out empty grasps. Defining grasps with rewards greater than 0.5 as successful, the system exhibits an almost perfect success rate upon convergence and even the intermediary performance was good yielding an average of 41 successful grasps out of the 60 trials. The system converged to learning two distinct styles of grasps, i.e., a robust grasp of the side of the paddle shown in Figure 3a and a handle grasp that is appropriate for the Barrett hand, shown in Figure 3b. Note that the discovered handle grasps have a distinctively unusual grasping style, which was discovered by the robot system.

The successful grasps superimposed onto the ECV descriptors of the object can be seen in Figure 4. For the human demonstrator, it was logical to attempt grasping the racket at the middle of the paddle (e.g., see Figure 4a). However, this grasp requires a higher level of accuracy as the initial gap between the fingers is about the same size as the diameter of the paddle. While they do work sometimes, the active learning will eliminate them due to their lack of robustness as can be observed in Figure 4b.



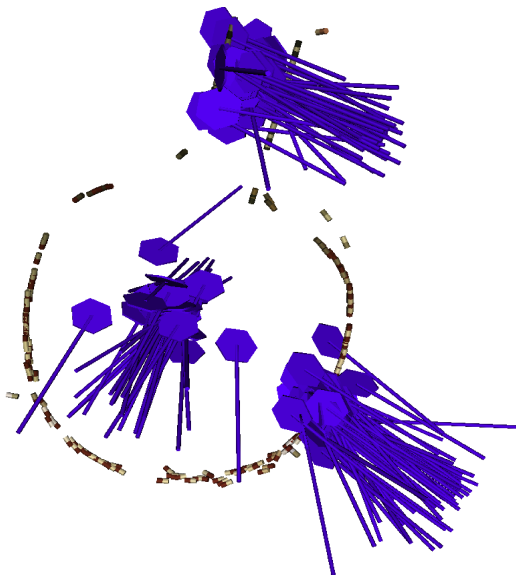
(a) Side Grasp



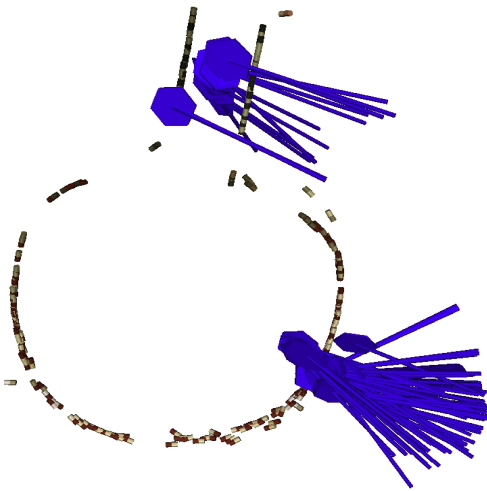
(b) Handle Grasp

Fig. 3: (a) The top figure shows the hand as it grasps the side of the paddle using a three fingered precision grip. (b) The bottom figure displays the grasp of the handle using a two fingered grasp by pressing the handle stably against the palm. Note that the end of the handle can just be seen at the tip of the middle finger.

The handle of the table tennis racket is nearly too small for the rather large and clumsy Barrett hand, allowing the hand to fully close about it as if it were an empty grasp. The handle also has a slightly cylindrical form, making a precision grip unstable. As a result, the human demonstrations for the handle were largely failures and the active learning system was expected to eliminate this grasp as well. Surprisingly, the grasp for the handle changed during active learning from imitation to a new one. Instead of the precision pinch implied by Figure 4a, the final grasp learned by the active learning system is of a new grasp type as shown in Figure 3b. The grasp employed uses only two fingers and presses the handle stably against the palm with properly placed finger tips; thus, the grasp no longer suffers from the stability problems of the precision pinch. Aligning the handle with the finger tips ensures that the fingers are blocked from fully closing and thereby prohibiting the false empty grasps which a power grasp approach would create. In order to achieve the later two-finger grasp, the system realized



(a) After Imitation Learning (Demonstrated Grasps)



(b) After Active Learning (Discovered Successful Grasps)

Fig. 4: (a) The top figure shows palm poses suggested by imitation learning superimposed onto the ECV descriptors of the racket. (b) The bottom figure shows successful palm poses suggested by the system after active learning.

a particularly stable strategy which enables the high success rate for the unusual and unintuitive grasp.

The system has been shown to be capable of learning quite complex grasps, beyond those demonstrated, that work reliably. These results are very promising.

IV. CONCLUSIONS

As shown in Section III-B, the results of the four experiments were positive as the system achieved confidently high success rates within less than 60 trials with learning taking place on the real system. The algorithm converged to a successful grasping policy, and found two distinct successful types of grasps. This fast learning performance indicates that the robot could potentially learn to grasp an object while

performing a pick and place task. The results also showed that the method has a lot of potential for finding unusual and complex grasps beyond those commonly suggested by the literature. The robot performed a considerable amount of exploration before and after it had found good grasps. The algorithm quickly determined which areas led to failed grasps, and focused an increasing amount of time on other areas. The system has shown that it can be used to refine demonstrated grasps to those suited to the specific robot hardware. Such a system could therefore be used to deal with the correspondence problem. The algorithm fulfilled all of its requirements.

In the future, more efforts will be needed in order to determine how the hyper-parameters, such as number of prior points and τ , affect the exploration-exploitation tradeoff. Faster convergence could also be achieved by using more complex reward functions, i.e., by doing reward shaping.

REFERENCES

- [1] A. Bicchi and V. Kumar. Robotic grasping and contact: a review. In *ICRA 2000 proceedings*, 2000.
- [2] D. Comaniciu and Peter Meer. Mean shift: A robust approach toward feature space analysis. In *Transactions on Pattern Analysis and Machine Intelligence*, 2002.
- [3] R. Detry, O. Kroemer, M. Popovic, Y.P.Touati, Emre Baseski, Norbert Krueger, J. Peters, and J. Piater. Object-specific grasp affordance densities. In *Proceedings of ICDL*, 2009.
- [4] R. Detry, N. Pugeault, and J. Piater. Probabilistic pose recovery using learned hierarchical object models. In *International Cognitive Vision Workshop*, 2008.
- [5] R.I. Hartley and A Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000.
- [6] M. Johnson and Y. Demiris. Abstraction in recognition to solve the correspondence problem for robot imitation. In *TAROS proceedings*, 2004.
- [7] N. Krueger, M. Lappe, and F. Woergoetter. Biologically motivated multimodal processing of visual primitives. *The Interdisciplinary Journal of Artificial Intelligence and the Simulation of Behaviour*, 2004.
- [8] Ruben Martinez-Cantin. *Active Map Learning for Robots: Insights into Statistical Consistency*. PhD thesis, University of Zaragoza, 2008.
- [9] M. Mason and J. Salisbury. *Robot Hands and the Mechanics of Manipulation*. MIT Press, 1985.
- [10] A. Morales, E. Chinelalto, P.J. Sanz, A.P. Pobil, and A. H. Fagg. Learning to predict grasp reliability for a multifinger robot hand by using visual features. In *AISC proceedings*, 2004.
- [11] M. A. Moussa and M. S. Kamel. A connectionist model of human grasps and its application to robot grasping. In *Neural Networks 1995 proceedings*, 1995.
- [12] J Pauli. Learning to recognise and grasp objects. *Machine Learning*, 1998.
- [13] R. Pelossof, A. Miller, P. Allen, and T. Jebara. An svm learning approach to robotic grasping. In *ICRA 2004 proceedings*, 2004.
- [14] N. Pugeault. *Early Cognitive Vision: Feedback Mechanisms for the Disambiguation of Early Visual Representation*. Vdm Verlag Dr. Mueller, 2008.
- [15] C.E. Rasmussen and C.K.I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.
- [16] M. Salganicoff, L.H. Ungar, and R. Bajcsy. Active learning for vision-based robot grasping. *Machine Learning*, 1996.
- [17] A. Saxena, J Dreimeyer, J. Kearns, C. Osondu, and A. Ng. *Experimental Robotics*, chapter Learning to Grasp Novel Objects using Vision. Springer Berlin, 2008.
- [18] A. Saxena, L.L.S. Wong, and A.Y. Ng. Learning grasp strategies with partial shape information. In *AAAI 2008 proceedings*, 2008.
- [19] J. Steffan, R. Haschke, and H. Ritter. Experience-based and tactile-driven dynamic grasp control. In *IRS proceedings*, 2007.
- [20] J. Steffan, R. Haschke, and H. Ritter. Som-based experience representation for dexterous grasping. In *International Workshop on Self-Organizing Maps*, 2007.