Learning Visual Features that Predict Grasp Type and Location

Di Wang and Andrew H. Fagg

***Abstract*— J. J. Gibson suggested that objects in our environment can be represented by an agent in terms of the types of actions that the agent may perform on or with the object. This *affordance* representation allows the agent to make a connection between the perception of key properties of an object and these actions. In this paper, we explore the automatic construction of visual representations that are associated with components of objects that afford certain types of grasping actions. A training data set of images is labeled with regions corresponding to locations at which certain grasp types could be applied to the object. A classifier is trained to predict whether particular image pixels correspond to these grasp regions. Each pixel that is classified as a positive example of a grasp region votes for its surrounding image region. If there exists a pixel with a large enough number of votes, then the image is considered to afford the grasp and the location of the pixel is identified as the best grasp point. Experimental results show that the approach is capable of identifying the occurrence of both handle-type and ball-type grasp options in images containing novel objects.**

I. INTRODUCTION

A robot faced with manipulating objects in the environment must be able to use visual information to identify the set of grasping and manipulation actions that are afforded by the objects contained therein [4]. This set of options not only helps the robot to plan the next sequence of actions to execute, but once the next action is selected, this representation can provide detailed information about the necessary shape and pose of the hand, and the forces to be applied. While information from haptic feedback will play a key role in this process [2], [9], the initial choices will often be made based on visual information only. What visual representations support the recognition of grasping actions and how can an agent automatically acquire these representations either by direct interaction with the environment or by observation of the actions of other agents?

Piater and Grupen [8] use a representation based on constellations of 2D appearance features to choose a particular grasp and to position the hand. The constellations of edge and texture features are learned as the robot haptically explores and grasps objects. Particular constellations that are predictive of successful grasps are cached for use in later recognition problems.

Saxena et al. [10] automatically acquire appearance-based visual models that can be used to label image regions as graspable or not using a precision type grasp. Given a large set of example images with specific graspable regions labeled, they employ a supervised learning approach to construct the pixel classifier. Given a labeling of multiple images of the same scene, 3D grasp locations are then estimated. Their experiments on a physical robot show promising results that novel objects can often be grasped even in cluttered scenes without pre-defined 3D models of the objects.

Following this idea, a possible next step is to explicitly identify locations in an image that afford specific types of grasps. These grasp types are defined, in part, by the hand shape and by the set of forces to be applied to the object. For example, a mug may be grasped by its handle (e.g., for the purposes of transporting or drinking from) or by its rim (e.g., for transporting or throwing). In this paper, we explore the problem of learning such a set of visual models based on a training set with a small number of objects and with a very coarse labeling of the regions in which specific grasps may be used. Our approach is to first extract high-dimensional feature vectors that describe the local shape and texture around points of interest. In particular, we use the scale invariant feature transform (SIFT) [5], which yields discriminative features that are robust to orientation, scale and lighting changes. We then use a classifier to identify those feature vectors that reliably make predictions about the grasp types and locations in the image. If a sufficient number of positively classified feature vectors is found within some part of an novel image, it is considered to afford that particular grasp type at the region in which the positive features are found. We demonstrate that our approach is capable of extracting visual representations that capture both handle-type grasps and ball-type grasps that are applied to cylindrical objects.

II. METHODS

Our goal is to visually identify the types of grasps that are applicable to a given object and, for each grasp, identify the approximate location at which grasp contacts are likely to occur. We would like for the learned visual representations to be generalizable to novel objects. Hence, the visual representations must be able to capture aspects of the object's shape that are relevant to the grasp. Because we would not like to commit *a priori* to particular, high-level visual primitives, we choose instead to make use of a general set of multi-scale, rotation invariant visual features. *SIFT* [5] features are capable of identifying fine details, including textures. This type of visual feature is considered to be very discriminative [6] and is widely used in object recognition related tasks. The SIFT approach identifies certain points in the image as being salient. This allows a recognition algorithm to focus on a small subset of the image pixels.

D. Wang is a Ph.D. student and University of Oklahoma Foundation Fellow, University of Oklahoma, Norman, OK 73019, USA di@cs.ou.edu

A. H. Fagg is an Associate Professor of Computer Science and Bioengineering, University of Oklahoma, Norman, OK 73019, USA fagg@cs.ou.edu

When applied to *maximally stable extremal regions (MSERs)*, SIFT features capture aspects of the gross shape of an object or its parts [3].

For each type of grasp to be recognized, each training data set image is labeled with one or more *grasp regions* at which the corresponding grasp could be applied. We first train a model that classifies SIFT feature vectors as to whether they are expected to fall within this grasp region. For a novel image, the classifier labels each candidate feature as a positive or negative example of the grasp region. Positive features "vote" for the region of the image in which they are found. SIFT features derived from the original image and from a MSER capture fundamentally different image properties. If there exists a pixel with a large enough number of votes from both types of features, then the image is considered to afford the grasp and the location of the pixel is identified as the best grasp point.

A. Visual Features

In this section, we briefly outline the SIFT and MSER-SIFT[1] approaches to describing image appearance.

1) SIFT: **Keypoint detection.** The goal for keypoint detection is to detect the locations in a given image that are robustly identifiable in other images of similar perspectives of the same or similar objects. Specifically, SIFT aims to identify keypoints that are repeatable when scaling and in-image rotation exist. First, scale space images are generated from the original image. Then, the Difference of Gaussian (DOG) responses are computed from the two nearby scales in this scale space. These DOGs respond highest to areas of high contrast (dark surrounded by bright, or vice-versa). The most stable locations are identified by the spatial and scale extrema in the DOG images. These stable locations are termed "keypoints" by Lowe [5]. The gradient (orientation) of a pixel in an image is defined as the direction in which the image intensity changes most quickly. A canonical orientation is assigned to each keypoint by using the dominant orientation of pixels within a surrounding region. In order to achieve rotation invariance, the orientation of the surrounding features are relativized to this canonical orientation.

SIFT descriptors. Each keypoint corresponds to a 2D location in a given image, which itself can be considered as defining the origin of a 2D coordinate frame, whose orientation and scale are determined by the canonical orientation and scale of the keypoint. The keypoint descriptor is a vector that represents the appearance of this patch. First, a patch is divided into subregions, and for each subregion, a histogram of local orientations is computed. Each bin of this histogram counts the number of pixels with gradients in a particular range of orientations. In practice, a shift by an individual pixel in either direction does not substantially change the orientation histograms. This property makes SIFT less sensitive to variations in registration of the patch location during the matching process. Finally, the values of all bins for each histogram are appended together into a single feature vector. Lowe shows experimentally that a descriptor of length 128 gives the best performance for feature matching. This corresponds to 4×4 patches and 8 bins in each patch. In order to reduce the effect of illumination changes, the descriptor vector is normalized to unit length.

2) MSER-SIFT: This approach uses an affine invariant shape descriptor for MSERs. A set of binary images can be generated by applying different thresholds to the original image. The set of extremal regions are the connected black or white regions in these binary images, and MSERs correspond to the regions that are stable across a set of thresholds. These MSERs are used to define SIFT keypoints. Instead of using grey-scale images to calculate feature vectors as in SIFT, MSER-SIFT uses the binary MSER itself to calculate the SIFT descriptors. Since MSERs usually capture the shape information of an image patch, the SIFT features calculated on these regions depend more on the shape of the patch rather than its texture. The detailed implementation is described by Forssén and Lowe [3].

B. Feature Vector Classification

The next problem is to classify a feature vector as to whether it comes from a region corresponding to a particular grasp type. We use a support vector machine (SVM) classifier [11] to solve this problem. Given two sets of n-dimensional vectors ($n = 128$ in our case), a SVM classifier attempts to construct a hyperplane that optimally separates the positive examples from the negative ones by giving the smallest classification error. In practice, we adjust the relative weighting between the positive and negative classes based on the number of samples in each class. By using a polynomial kernel of degree k (larger than 1), the SVM classifier will search for a hyperplane in a higher dimensional space in which the feature vector describes all possible combinations of k-degree products of the original 128 feature elements. While the discrimination power is dramatically increased, the computational complexity is not significantly increased by using a *kernel trick* [11], which allows the computations to remain in the original 128-dimensional space.

C. Identifying the Grasp Region

In practice, individual positive keypoint feature vectors can be located in many locations across an image containing an object. However, a larger number of positive features are typically found in image areas that contain the target concept. We combine evidence from all observed features in an image by using a non-parametric particle based voting approach [12]. Features that are recognized as positive vote for surrounding pixels using a Gaussian-shaped voting function. The width of this function is proportional to the feature's scale. Also, the overall strength of the vote of an individual feature is scaled by the degree to which it matches the expected scale.

More formally, we define the voting function for a partic-

[1] Informally, we use MSER-SIFT or MSER feature to denote the shape descriptor computed on a MSER using SIFT.

ular grasp type at location x in an image as:

$$V(x) = \sum_{i=1}^{N_p} W_i(x), \qquad (1)$$

where N_p is the total number of positive features and $W_i(x)$ is a Gaussian-shaped weighting function for each positive feature i. We define the weighting function for feature i as:

$$W_i(x) = \alpha_i e^{-\frac{(x-x_i)^T(x-x_i)}{2\sigma_i^2}}, \qquad (2)$$

where x denotes an arbitrary location in the image, x_i denotes the location of positive feature i, σ_i determines the width and α_i determines the magnitude. Specifically, we define:

$$\sigma_i^2 = \beta s_i^2, \qquad (3)$$

where s_i denotes the scale of feature i. For SIFT features, we directly use the scale of the image in scale space; for MSER-SIFT features, we use the mean of the major and minor axes of the bounding ellipse. $\beta = 0.5$ is an empirically selected constant. The magnitude of (2) is determined by α_i, whose value is highest if feature i has a scale that is expected given the training set:

$$\alpha_i = e^{-\frac{(s_i-\bar{s}_i)^2}{2\hat{\sigma}_i^2}}, \qquad (4)$$

where $\hat{\sigma}_i$ and $\bar{s}_i$ are the standard deviation and the mean of the k positive features in the training set that have the smallest Euclidean distance to feature i.

D. Combining SIFT and MSER-SIFT Features

In practice, SIFT features are very specific to the local appearance of a region. As a consequence, they can often be "distracted" by surface textures. MSER-SIFT features allow our algorithm to focus more on the shape of objects rather than textures. However, many fewer MSER-SIFT features are typically found than SIFT features. Our approach is to combine information from both sources of evidence. Specifically, we combine the values of the particle voting functions of SIFT and MSER-SIFT features by a pixel-wise multiplication:

$$V_{\text{both}}(x) = V_{\text{SIFT}}(x) \times V_{\text{MSER}}(x). \qquad (5)$$

E. Identifying a Grasp

We consider an image as affording a particular grasp if some pixel accumulates enough votes. Specifically, if $\max_x V_*(x) > \theta_*$, then we assume that a grasp has been identified at location x, where $* \in \{\text{SIFT, MSER, both}\}$ and θ_* is a threshold. We select the threshold that gives the Kolmogorov-Smirnoff distance between the true positive rate (TPR = TP/(TP+FN); TP = true positives; FN = false negatives) and false positive rate (FPR = FP/(FP+TN); FP = false positives; TN = true negatives) of the training set [13].

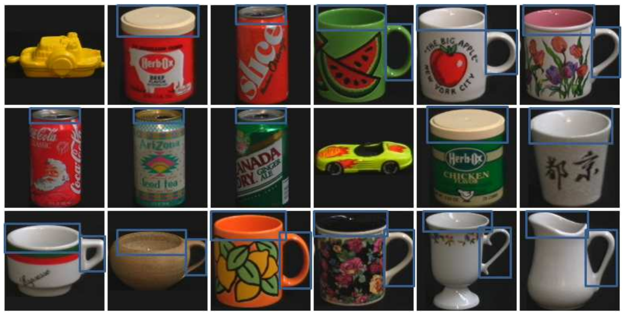


Fig. 1. The set of objects used in our experiments. The rectangular region(s) in each image corresponds to the grasp region, which is manually selected.

III. EXPERIMENTAL RESULTS

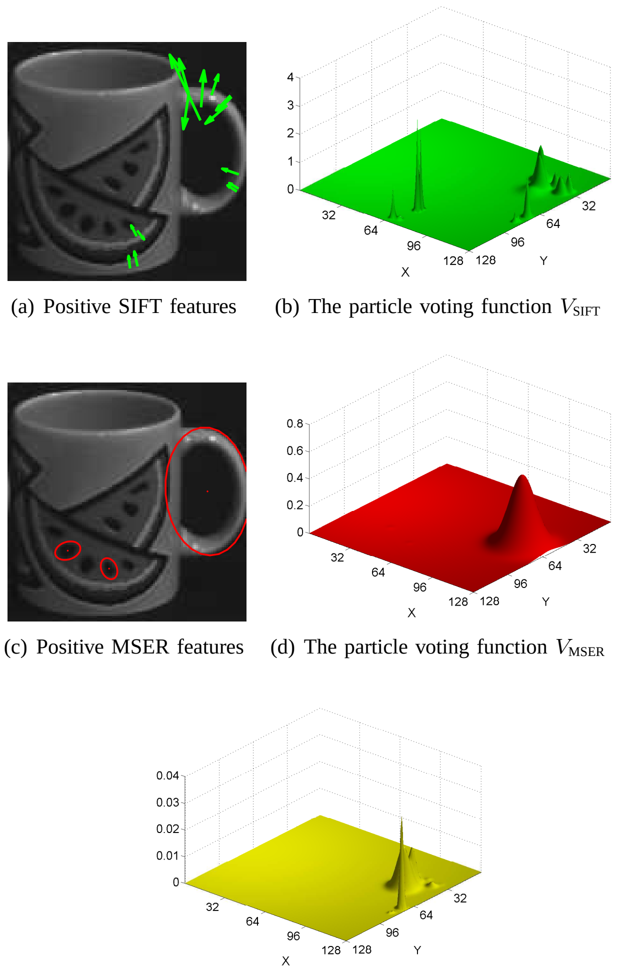
For the purposes of examining the behavior of our algorithm, we choose to focus on two grasps: a ball grasp from the top of circular-shaped objects and a handle grasp for mug-shaped objects. Based on this, we selected 18 objects from the COIL-100 database [7], which includes all 9 mugs, 7 can-shaped objects (including all the soda cans) and 2 other objects (a ship and a car), as shown in Fig. 1. The rectangular region(s) shown in each image corresponds to the *grasp region* that is manually selected by the author. This grasp region is used in the training set for the purposes of labeling features as being positive or negative examples of a grasp region. In addition, this region is used in the evaluation phase of our experiments. For each object, we use all 72 images which correspond to all viewing angles surrounding the object with 5 degree increments.

We use VLFeat [14] to calculate SIFT features and MSER keypoint locations. We use Forssén and Lowe's algorithm [3] to calculate MSER-SIFT descriptors. Given this relatively small set of objects, we use leave-one-object-out cross-validation to evaluate the performance of our algorithm. The features within a grasp region for a particular grasp are labeled as positive, otherwise they are labeled as negative.

A. Performance Effect of Feature Type

For a selected testing object, we use the other 17 objects to construct a model for each of the two grasp types. For each grasp type, we train two separate classifiers: one for SIFT and the other for MSER-SIFT features. We use a c-svc (support vector classifier) with a polynomial kernel of degree 10 provided by the LIBSVM Matlab toolbox [1]. We choose the relative cost weighting of the two classes to be inversely proportional to the number of features in each class as observed in the training set.

Examples of SIFT features identified in a test image are shown in Fig. 2(a). The tail of each arrow denotes the center (keypoint) and the length denotes the scale of each SIFT feature. Examples of MSER-SIFT features that are recognized as positive are shown in Fig. 2(c). Each ellipse denotes a MSER-SIFT feature, which is scaled based on the actual feature scale. The particle voting functions calculated from the SIFT and MSER features are shown in 3D in Fig. 2(b) and Fig. 2(d). The origin of these figures



(a) Positive SIFT features (b) The particle voting function V_{SIFT}

(c) Positive MSER features (d) The particle voting function V_{MSER}

(e) The particle voting function V_{both}

Fig. 2. Predict grasp type and location in an image containing a mug. The origin of the 3D coordinate system corresponds to the lower left corner of the test image.

corresponds to the lower left corner of the test image. We can see that the locations of the peaks correspond to the locations of the positively classified keypoints. Note that the two small MSER features in Fig. 2(c) are assigned nearly zero height in Fig. 2(d). This is due to the fact that the scales of these two features are very different from those of the most similar features in the training set. The pixel-wise multiplication of these two particle voting functions is shown in Fig. 2(e). By using this "and" operation, the MSER-SIFT votes filter out the extraneous SIFT votes and the SIFT votes refine the locations that are supported by the MSER-SIFT votes. We select the peak in Fig. 2(e) as the predicted grasp location as long as it is higher than the threshold, θ_{both}.

We compare the grasp prediction performance of three algorithms: SIFT feature only, MSER feature only and the combination of both features. SIFT-only and MSER-only approaches only use votes of a single type of feature (V_{SIFT} or V_{MSER}), while the combined approach uses the product of the two (V_{both}). We use two criteria for performance evaluation. First, we evaluate whether a certain type of grasp can be identified correctly for a given image (a yes/no criterion). We report the performance of a particular algorithm on a specific grasp using a single contingency table and summarize the performance using the kappa statistic:

$$\kappa = \frac{\text{observed agreement} - \text{chance agreement}}{1 - \text{chance agreement}}. \tag{6}$$

The kappa statistic measures performance of the grasp recognition models relative to the best strategy possible without any information. $\kappa \leq 0$ is interpreted as performance being no better than a fixed strategy; $\kappa = 1$ is interpreted as having perfect performance. When a grasp type is correctly identified (a true positive), the next question is whether the predicted grasp location is within the correct region (a precision criterion). We calculate the percentage of correct locations (PCL) for the true positive images only.

The contingency tables of the SIFT-only approach, MSER-only approach and the combined approach for the handle grasp are shown in Tables I-III. For the handle grasp, the SIFT-only feature approach performs poorly at recognizing negative examples (many false positives). Because SIFT features are sensitive to fine details of texture, it is not uncommon for a positive SIFT feature to be observed in a negative image. The MSER-only feature approach correctly recognizes many negative cases (true negatives). However, the number of false negatives is also high. This is because that MSER features depend more on the gross shape of an object. The positive images in our testing set usually contain a single large scale MSER feature enclosing the handle (such as the one observed in Fig. 2(c)), which is rarely observed in the negative images without handles. The combined approach takes advantages of both features and works even better on the negative examples, since only locations voted for by both features are identified as grasp points. However, this comes at a cost of a larger number of false negatives than either of the other two approaches. By comparing the performance of these three approaches, we can see that the combined feature approach has the highest κ. Note that the combined feature approach has the lowest number of true positives. However, of the images that it does correctly label as positive, it also correctly identifies the grasp region in the largest proportion as compared with the other approaches. This is seen in the higher PCL for the combined approach.

For the top grasp, unlike the handle grasp, there are only two negative objects: the car and the ship. The labels for all testing images are positive except for the images that correspond to these two objects. The corresponding results are shown in Tables IV-VI. The SIFT-only feature approach simply identifies all images as positive and performs equally to the chance agreement ($\kappa = 0$). This is due to the ambiguity of textures and a low threshold. However, the PCL of SIFT-only approach is high. This is because that the SIFT features are very discriminative and the positive features usually aggregate in the correct grasp regions. In contrast, MSER-only feature approach correctly recognizes most of the negative images (114/144). This is because the gross shapes of the car and the ship are very different from

TABLE I

HANDLE GRASP PREDICTION RESULT: SIFT

κ=0.0286		Actual		
PCL=63.89%		P	N	Total
Predicted	P	529	738	1267
	N	1	28	29
	Total	530	766	1296

TABLE II

HANDLE GRASP PREDICTION RESULT: MSER

κ=0.3231		Actual		
PCL=66.14%		P	N	Total
Predicted	P	378	289	667
	N	152	477	629
	Total	530	766	1296

TABLE III

HANDLE GRASP PREDICTION RESULT: BOTH

κ=0.4975		Actual		
PCL=96.06%		P	N	Total
Predicted	P	279	45	324
	N	251	721	972
	Total	530	766	1296

TABLE IV

TOP GRASP PREDICTION RESULT: SIFT

κ=0		Actual		
PCL=82.99%		P	N	Total
Predicted	P	1152	144	1296
	N	0	0	0
	Total	1152	144	1296

TABLE V

TOP GRASP PREDICTION RESULT: MSER

κ=0.1171		Actual		
PCL=72.30%		P	N	Total
Predicted	P	592	30	622
	N	560	114	674
	Total	1152	144	1296

TABLE VI

TOP GRASP PREDICTION RESULT: BOTH

κ=0.2259		Actual		
PCL=90.63%		P	N	Total
Predicted	P	811	39	850
	N	341	105	446
	Total	1152	144	1296

the positive objects where a top grasp can be applied. The combined feature approach gives the highest κ and PCL. In this particular case, an algorithm that guesses all images as positive will perform well. Hence, there is very little room for improvement. This explains in part why $kappa$ for all three approaches is low.

B. Performance Effect of Training Set Size

In the second set of experiments, we are interested in the role that training set size plays in algorithm performance. We examine this issue by varying the number of mugs in the training set while keeping the 9 non-mug objects constant. First, we fix the order of the 9 mugs randomly. Then, for each of the 9 mugs we selected, we add between 1 and 8 mugs to the training set according to this order. This gives a training set size of 10 to 17. The 9th mug is used for testing. For the handle grasp, we expect the performance to increase substantially as mugs are added to the training set, since there are no positive examples in the constant part of the training set. However, for the top grasp, we only expect a small performance increase with a relatively high starting point, since most of the objects in the constant training set are positive examples (the can-shaped objects) of the top ball grasp.

For the handle grasp, the handle of a mug is visible in 13 out of the 72 aspects on average; for the top grasp, all mug images are positive examples since the top of a mug can always be observed. Because the chance agreement is very high in this case, the kappa statistic is degenerate. Therefore, we report the true positive rate (TPR) instead.

Fig. 3 shows the mean TPRs of 9 mugs as the number of mugs in the training set increases. In this figure, we can see that both handle grasp and top grasp performance improve with increasing training set size. The large standard deviations are due to the two mugs with low image qualities in our data set (the second and the fourth mugs in the last row in Fig. 1). Also, top grasps overall have higher TPRs than handle grasps. This is because that there are more positive examples (the can-shaped objects) with top grasps in the training set, which helps our algorithm to learn a better model. For both the handle grasp and the top grasp, there is a significant difference in TPR between 2 mugs and 8 mugs in the training set ($p < 10^{-3}$ for the handle grasp; $p < 0.03$ for the top grasp, according to a paired bootstrap test).

Likewise, the PCLs are shown in Fig. 4. Note that the first point of the blue curve is a mean of only 5 trials, given that grasps are only successfully identified for 5 of the 9 test mugs. All the other points are means of 9 mugs. In this figure, we can see that handle grasps have much smaller standard deviations and overall higher values than top grasps. This is due to the fact that visual features corresponding to handle grasps are usually more distinctive and reliable than top grasps. Both curves show little variation as a function of training set size. For both the handle grasp and the top grasp, there is no significant difference between 2 mugs and 8 mugs in the training set ($p < 0.3$ for the handle grasp; $p < 0.8$ for the top grasp, according to a paired bootstrap test).

IV. CONCLUSIONS AND FUTURE WORK

In this paper, we propose an approach that visually identifies grasp types and locations from images of novel objects. For a given image, visual features are classified based on whether they correspond to a particular grasp type. This classifier is trained using images with manually labeled grasp regions. In a novel image, the evidence from multiple identified features is combined using a particle

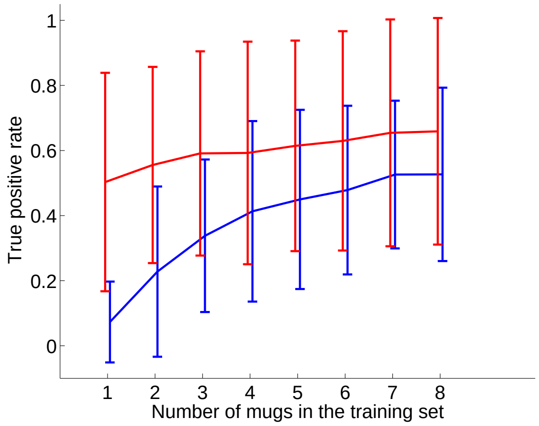


Fig. 3. The mean and standard deviation (shown as whiskers) of true positive rates as the number of mugs in the training set increases. Blue: handle grasp; red: top grasp.

voting function. The peak location in an image is selected as the grasp location if its value exceeds a threshold.

Our experiments show that the learned model can be generalized to novel objects with similar shapes as the ones in the training set. The combined approach that uses both SIFT (texture-sensitive) and MSER-SIFT (shape-sensitive) features improves the performance substantially. Increasing the number of positive examples in the training set increases the true positive rate for grasp identification. However, as long as a certain grasp type is correctly identified, the accuracy of the location tends to remain the same. This reflects the overall performance of our particle voting function, that is, the peaks usually fall within the correct grasp regions.

Our algorithm shows the potential to grasp novel objects, as long as these objects share similar components as the ones in the training set. For several cases, however, our algorithm exhibited poor performance in properly classifying certain types of images. This effect is due in part to a poor choice of threshold (a low value) for the voting function. We believe that this can be corrected in part by introducing a separate validation data set with which this threshold can be selected (as opposed to using the training data set). However, an increase in this threshold will yield a higher false positive rate in trade for a lower false negative rate. In part, we believe that this can be compensated for by using a larger number of objects for training and validation. In addition, we expect that a robot that fails to properly identify a grasp option from a single image will have other opportunities as more images are available in the on-line context (e.g., through stereo pairs or images taken in time).

As future work, we plan to generalize our experiment in several ways. First, we plan to use images directly taken from real scenes, which will include cluttering and occlusion. Second, we currently manually label the grasp regions in a given image. This process may become cumbersome when the number of objects in the data set increases. We plan to automatically label grasp regions by observing how human teachers place contacts on objects.

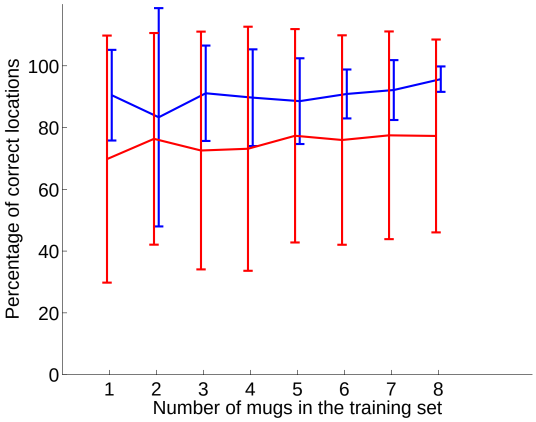


Fig. 4. The mean and standard deviation (shown as whiskers) of the percentage of correct locations as the number of mugs in the training set increases. Blue: handle grasp; red: top grasp.

V. ACKNOWLEDGMENTS

This work is supported in part by the Foundation Fellowship Grant provided by the Graduate College of the University of Oklahoma.

REFERENCES

[1] C.-C. Chang and C.-J. Lin. *LIBSVM: a library for support vector machines*, 2001. Software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm.

[2] J. A. Coelho and R. A. Grupen. Constructing effective multifingered grasp controllers. In *Proceedings of the 1994 Conference on Robotics and Automation*, San Diego, CA, May 1994. IEEE.

[3] P.-E. Forssén and D. G. Lowe. Shape descriptors for maximally stable extremal regions. In *IEEE International Conference on Computer Vision*, volume CFP07198-CDR, Rio de Janeiro, Brazil, October 2007. IEEE Computer Society.

[4] J. J. Gibson. The theory of affordances. In R. E. Shaw and J. Bransford, editors, *Perceiving, Acting, and Knowing*. Lawrence Erlbaum, Hillsdale, 1977.

[5] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[6] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *IEEE Trans. Pattern Anal. Mach. Intell.*, 27(10):1615–1630, October 2005.

[7] S. A. Nene, S. K. Nayar, and H. Murase. Columbia object image library: COIL-100. Technical Report CUCS-006-96, Department of Computer Science, Columbia University, February 1996.

[8] J. H. Piater and R. A. Grupen. Learning appearance features to support robotic manipulation. In *Proceedings of the Cognitive Vision Workshop*, 2002. Electronically published.

[9] R. Platt, Jr., A. H. Fagg, and R. A. Grupen. Nullspace composition of control laws for grasping. In *Proceedings of the International Conference on Intelligent Robots and Systems*, pages 1717–1723, 2002.

[10] A. Saxena, J. Driemeyer, and A. Y. Ng. Robotic Grasping of Novel Objects using Vision. *The International Journal of Robotics Research*, 27(2):157–171, 2008.

[11] B. Scholköpf and A. J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, Cambridge, MA, USA, 2001.

[12] S. Thrun, W. Burgard, and D. Fox. *Probabilistic robotics*. Intelligent robotics and autonomous agents. MIT Press, Cambridge, Massachusetts, 2005.

[13] P. E. Utgoff and J. A. Clouse. A kolmogorov-smirnoff metric for decision tree induction. Technical Report Computer Science Technical Report 96-3, University of Massachusetts, Amherst, 1996.

[14] A. Vedaldi and B. Fulkerson. VLFeat: An open and portable library of computer vision algorithms. http://www.vlfeat.org/, 2008.