Cognitive Vision Workshop, 19-20 September 2002, Zürich, Switzerland.

Learning Appearance Features to Support Robotic Manipulation

Justus H. Piater
Projet PRIMA, Laboratoire GRAVIR-IMAG
INRIA Rhône-Alpes, 655 av. de l'Europe, 38330 Montbonnot, France

Roderic A. Grupen
University of Massachusetts, Amherst, MA 01003, USA

Abstract

A predominant purpose of vision is to facilitate interaction with the world. Motivated by biological vision, we argue that many common vision-guided activities can be carried out directly on the basis of features of appearance, and do not require elaborate world models. We describe a visual feature learning system that supports a haptically-guided, dextrous robotic grasping system. It learns features, combinations of Gaussian-derivative filter responses, that correlate well with successful grasping parameters. Without explicit knowledge of object identities or categories, the system learns to propose object-specific grasp parameters, considerably improving the quality of haptically-guided grasps with respect to the "blind" system. The combined system is loosely anthropomorphic in that it is guided by vision for hand pre-shaping, and by haptics during execution of a grasp, without explicit object recognition, scene reconstruction, or path planning.

1 Vision For Action

From the perspective of biological and artificial autonomous systems, the purpose of vision is to facilitate interaction with the world. It has long been known that the relatively primitive neural systems of many animals are highly effective at closing the loop between perception and action. A frequently-cited example is the vision-guided behavior of frogs that is controlled by several distinct and hard-wired neural pathways [8], including retinal "bug detectors" [2]. The visual flight control of house flies is well described by a small set of differential equations [17].

Tasks such as these still pose challenges to computer vision research, even though they can evidently be performed by quite simple computational devices. It can safely be asserted that neither fly nor frog brains maintain elaborate world models. In contrast, most current research in computer vision is based on Marr's proposal that the immediate purpose of human vision is to build mental representations of the world [11]. However, recent research in psychology and neurophysiology has accumulated considerable evidence that even the human visual system contains distinct visual pathways dedicated to particular visual tasks, many of which do not involve elaborate representations of the world. In particular, the primary task of the dorsal visual stream (from the primary visual cortex into the posterior parietal cortex) appears to be visual control of motor behavior, with little or no involvement of the persistent visual representations formed by the ventral stream (into the inferotemporal cortex) [12].

Evidently, humans perform everyday manipulations to a large extent using model-free visuomotor control based on rather direct links from perception to action. This motivates research of vision as task-driven, interactive learning systems. In robotics research, the success of image-space visual servoing methods indicates the promise of appearance-based control [13, 9].

We describe a visual learning system that discovers visual features to support a haptically-guided robotic grasping system. By observing grasps, the visual system learns features of object appearance that robustly predict successful grasp parameters. Object-specific predictions are formed directly on the basis of observed features, without explicit object recognition or geometric interpretation, and are used to pre-shape the dextrous hand before haptic contact with the target object is made. The grasp itself is then performed using haptic feedback alone. We demonstrate that the expected quality of grasps is drastically improved as a result of this learning procedure.

2 Haptically-Guided Grasping

Coelho [4] describes a framework for closed-loop grasping using a dextrous robotic hand equipped with touch sensors at the fingertips. Reference and feedback error signals are expressed in terms of the residual forces and torques acting on the object. These are locally estimated using sensor feedback in the form of fingertip contact positions and normals. On the basis of error signals, the grasp controller computes incremental displacements for a subset of the contacts so as

We are indebted to Jefferson Coelho for his active support. This work was funded in part by the National Science Foundation under grants CISE/CDA-9703217, IRI-9704530 and IRI-9503687, and by the Air Force Research Labs, IFTD (via DARPA) under grant F30602-97-2-0032.

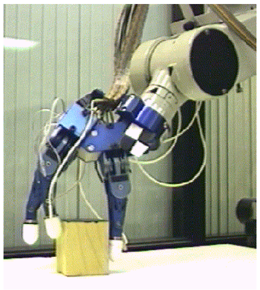

Figure 1. The Stanford/JPL dextrous hand performing haptically-guided closed-loop grasp synthesis.

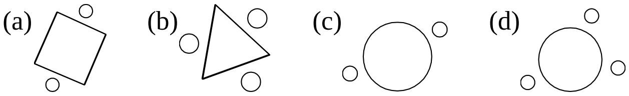

Figure 2. By Coelho's formulation, some objects are better grasped with two fingers (a), some with three (b), and for some it is unimportant (c, d).

to reduce the error. As a result, the fingers probe the surface of the object until their positions converge to a local minimum in the force/torque error surface.

The three-fingered Stanford/JPL robotic hand (Fig. 1) employed in this work can perform grasps using any two- or three-fingered grasp configuration. Each of these finger combinations c gives rise to an individual grasp controller π_c, yielding a total of four controllers. For the purpose of this paper, it is sufficient to distinguish between two classes of grasp controllers, namely, two- and three-fingered grasps.

The quality of a grasp is characterized by the minimum friction coefficient μ_0 required for the object to resist the residual forces exerted by the fingertips in their current positions. For an ideal grasp, $\mu_0 = 0$, meaning that the grasp configuration is suitable for fixing the object in place using frictionless point contacts. Depending on the object shape and the number of fingers available, this ideal may not be achievable (Fig. 2).

3 Vision For Hand Pre-Shaping

Coelho's haptically-guided grasping system begins each grasp with a random hand configuration. During execution, a grasp controller may end up in a local optimum and produce a poor grasp. One way to enhance the robustness of the grasps is to learn reactive policies for switching between individual grasp controllers π_c [5]. Instead, here the goal is to learn visual features that predict successful hand orientations, given prior haptic experience. These can be used to choose a two- or a three-fingered grasp controller, and to pre-shape the hand in order to place fingertip contacts reliably near high-quality grasp configurations for a task. From there, the chosen grasp controller can lead reliably to the preferred solution.

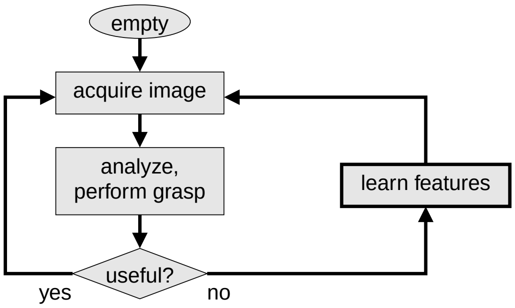

Figure 3. Scenario for learning features to propose grasp parameters.

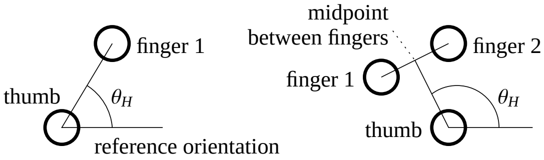

Figure 4. Definition of the hand orientation (azimuthal angle) for two- and three-fingered grasps.

The general scenario for the grasping problem is shown in Fig. 3. Before the onset of a grasp, the visual system acquires an overhead view of the target object. This image is searched for previously learned features that correlate with successful grasp parameters, including the relative hand orientation and the number of fingers used. On the basis of any such features found in the image, grasp parameters are proposed to the grasping system. Subsequently, the grasp is executed using these parameters. Upon convergence of the haptic controller to a final grasp configuration, grasp parameters are recorded and used for learning. The orientation θ_H of the hand during a given grasp configuration is defined as illustrated in Fig. 4.

The robot may encounter a variety of objects that differ in their shapes. Each object category may require dedicated features to propose a hand orientation. Object identities are known to neither the haptic nor the visual component; the need for category-specific features must be discovered by grasping experience. Visual features that respond selectively to haptic categories permit the system to choose the number of fingers to use for a grasp (Fig. 2).

4 Compositional Features

To incrementally learn features that predict object-specific grasp parameters, we require an open-ended feature set that can deliver increasingly specific features, as demanded by the evolving task. To date, such on-line visual learning scenarios have not received much attention by the computer vision research community. To address these requirements, we define an infinite feature space that describes appearance properties at various levels of specificity, and that permit the computation of their orientation. Earlier work employed a similar feature space to learn rotation-invariant object recognition [14, 15]. Here, these properties allow features to be constructed that respond selectively to specific object categories, and whose orientations correlate with those of the robotic hand. The feature space is defined by *primitive* features that can be combined into *compound* features. Primitive features are local appearance descriptors represented as vectors of local filter responses [16, 10, 18]. These filters are oriented derivatives of 2-D Gaussian functions

$$G(\mathbf{x}, \sigma) = \frac{1}{2\pi\sigma^2} \mathrm{e}^{-\frac{\mathbf{x}^T\mathbf{x}}{2\sigma^2}}. \tag{1}$$

The oriented derivative of order d at orientation θ is computed as

$$G_d^\theta(\mathbf{x}, \sigma) = \frac{\partial^d}{\partial^d x} G(R_\theta \mathbf{x}, \sigma) \tag{2}$$

with a rotation matrix

$$R_\theta = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}.$$

Specific orientations are chosen such that they form a steerable basis for a given derivative d [7]. Here, the steerability property permits the efficient computation of filter responses at any orientation, given $d + 1$ measured filter responses for the dth derivative at orientations spaced $\pi/(d+1)$ degrees apart. The orientation of such a feature is defined by the responses of the first-derivative operators:

$$\tan\theta = \frac{G_1^{\pi/2}}{G_1^0} \tag{3}$$

Gaussian-derivative features have a number of desirable properties and are therefore widely employed by feature-based techniques in computer vision. They have also been used to model receptive fields found in the mammalian early visual pathway [21].

Our system employs two types of primitive features: An *edgel* is defined by the two first-derivative Gaussian basis functions ($G_1^{\pi/2}$ and G_1^0), encoding the local intensity gradient. A *texel* is represented as a 15-vector containing the responses to the basis filters of the first two derivatives at three scales σ (2 first-derivative kernels plus 3 second-derivative kernels, times 3 scales = 15 filters total). This encodes a local texture signature [16].

Primitive features by themselves are not very discriminative. However, spatial combinations of these can express a wide range of appearance characteristics at various degrees of specificity or generality. Compound features are rigid geometric combinations of primitive features, defined by the angles and distances between them (Fig. 5). In the following discussion, $\mathbf{f}$ denotes a feature, that can be either a primitive edgel or texel, or a compound feature consisting of edgels and texels. A primitive feature is defined by the 2- or 15-vector of filter responses. A compound feature is defined by the concatenated filter response vectors of the constituent subfeatures, and by their spatial arrangement (Fig. 5).

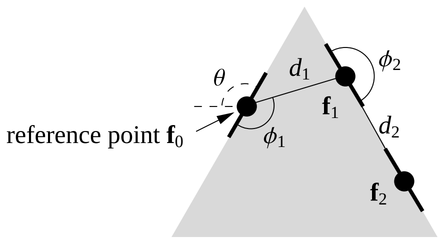

Figure 5. A geometric compound feature of order 3, composed of three primitives. The feature is defined by the angles ϕ and the distances d. Each primitive is either an edgel or a texel.

To measure the degree to which a prototype feature $\mathbf{f}$ is present at a pixel location $\mathbf{p}$, the applicable vector of filter responses, denoted $\mathbf{f}(\mathbf{p})$, is measured at $\mathbf{p}$. If $\mathbf{f}$ is a compound feature, then the designated reference point (Fig. 5) of the feature is placed at $\mathbf{p}$, and the filter responses are obtained at the respective locations corresponding to the spatial arrangement defined by $\mathbf{f}$. The *value* $f_\mathbf{f}(\mathbf{p}) \in [0, 1]$ of feature $\mathbf{f}$ at pixel location $\mathbf{p}$ is then computed by correlating the vector $\mathbf{f}(\mathbf{p})$ of filter responses at $\mathbf{p}$ with the response vector defining $\mathbf{f}$:

$$f_\mathbf{f}(\mathbf{p}) = \max\left\{0, \frac{\mathbf{f}^T\,\mathbf{f}(\mathbf{p})}{\|\mathbf{f}\|\,\|\mathbf{f}(\mathbf{p})\|}\right\} \tag{4}$$

The value of a feature in an image I is given by $f = \max_{\mathbf{p}\in I} f_\mathbf{f}(\mathbf{p})$. All feature values are computed in a rotationally-invariant manner by normalizing the filter responses with respect to orientation prior to correlation, using the steerability of Gaussian-derivative filters [16, 14]. The orientation of a primitive feature is given by Eqn. 3; that of a compound feature is defined by the orientation of the reference point.

5 Correlating Feature and Hand Orientations

Suppose we have a feature $\mathbf{f}$ whose image-plane orientation correlates well with the orientation of the robotic hand. In other words, its orientation $\theta_\mathbf{f}$ should be related to the azimuthal orientation θ_H of the robotic hand by a constant additive offset θ. A given feature, observed during many

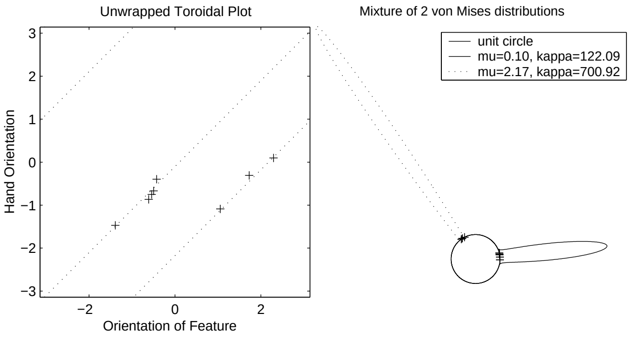

Figure 6. Left: Data points induced by a given feature on various images of an object form straight lines on a torus (two in this case); right: A mixture of two von Mises distributions was fitted to these data.

grasping experiences, generates data points that lie on a straight line on the toroidal surface spanned by the hand and the feature orientations (Fig. 6), described by $\theta_H = \theta_{\mathbf{f}} + \theta$. In fact, there may be more than one such straight line at different offsets θ because a given visual feature may respond to more than one specific object location (e.g., due to object symmetries), or to several distinct objects that differ in shape.

The remaining problem is to find the offsets θ associated with a feature. Assuming these can be modeled as random variables with unimodal and symmetric probability distributions $p_k(\theta)$, with the k corresponding to the clusters of points in Fig. 6, then this is a clustering problem in one-dimensional circular (angular) space, with the number of clusters K unknown, and can be represented as a mixture distribution

$$p_{\mathrm{mix}}(\theta) = \sum_{k=1}^{K} p_k(\theta)\, P(k) \tag{5}$$

with mixture proportions $0 < P(k) < 1$, $\sum_k P(k) = 1$.

One probability distribution that is often used in lieu of a normal distribution on a circular domain is the *von Mises* distribution [6]. It has the probability density function

$$p_{\mathrm{vM}}(\theta \mid \mu, \kappa) = \frac{\mathrm{e}^{\kappa \cos(\theta - \mu)}}{2\pi I_0(\kappa)} \tag{6}$$

where $I_0(\kappa)$ is the modified Bessel function of order zero. The mean direction of the distribution is given by μ, and κ is a concentration parameter with $\kappa = 0$ corresponding to a circular uniform distribution, and $\kappa \to \infty$ to a point distribution. In the right panel of Fig. 6, the probability density at an angle corresponds to the radial distance from the unit circle to the density curve.

Using the von Mises distribution, Eqn. 5 becomes

$$p_{\mathrm{mix}}(\theta \mid \mathbf{a}) = \sum_{k=1}^{K} p_{\mathrm{vM}}(\theta \mid \mu_k, \kappa_k)\, P(k) \tag{7}$$

where **a** is shorthand for the collection of parameters μ_k, κ_k and $P(k)$, $1 \leq k \leq K$. For all plausible numbers of clusters K, a $(3K - 1)$-dimensional non-linear optimization problem is solved to find the μ_k, κ_k and $P(k)$. The appropriate number K of mixture components can be chosen using any applicable method; as better features are learned, the clusters of data points will become increasingly well separated, making the problem easier. We use the Integrated Completed Likelihood criterion [3], an adaptation to clustering problems of the well-known Bayesian Information Criterion [19].

For a particular value of K, a maximum-a-posteriori (MAP) estimate of a mixture parameterization **a** can be computed using Bayes' rule. Here we assume a uniform prior probability density over all possible model parameterizations $\mathbf{a}_m$. In this case, the MAP estimate is identical to the maximum-likelihood estimate of **a** that maximizes the log-likelihood of the observed data $\Theta = \{\theta_1, \ldots, \theta_N\}$:

$$\log L(\Theta \mid \mathbf{a}) = \sum_{i=1}^{N} \log p_{\mathrm{mix}}(\theta_i \mid \mathbf{a}) \tag{8}$$

The θ_i are N angular offsets between feature and hand orientations that have been observed during actual grasps. This optimization problem can be solved by gradient descent on the partial derivatives of $\log L(\Theta \mid \mathbf{a})$, subject to the applicable constraints on the κ_k and $P(k)$.

6 Selecting Object-Specific Data Points

If different types of objects are encountered, dedicated features may have to be learned. Without a supervisor providing object identities, the data collections (Fig. 6) will be an indiscernible mix of feature responses corresponding to various objects, and any existing correlations of hand and feature orientations will be obscured. The key to learning dedicated features is to ignore data points corresponding to weak feature responses when computing the angular offsets as described in the preceding section. This permits features to emerge that respond strongly only to specific, highly characteristic object parts, but that respond weakly in any image that does not contain such an object. These weak responses will be ignored, and reliable models of θ can be fitted to the strong responses.

Deciding whether a given point is "strong" in this sense involves a threshold. Such thresholds $\alpha_{\mathbf{f}}$, specific to each feature, can be determined optimally in the Bayesian sense that the number of poor recommendations made by the resulting model is minimized. To do this for a given feature **f**, the history of feature responses $f_{\mathbf{f}}$ and the associated prediction errors $\Delta\theta_{\mathbf{f}}$ are analyzed in order to find a threshold $\alpha_{\mathbf{f}}$ such that most predictions for cases with $f_{\mathbf{f}} > \alpha_{\mathbf{f}}$ are correct. To formalize this intuitive notion, a global threshold $t_{\Delta\theta}$ is introduced, meaning that a prediction with $\Delta\theta_{\mathbf{f}} < t_{\Delta\theta}$

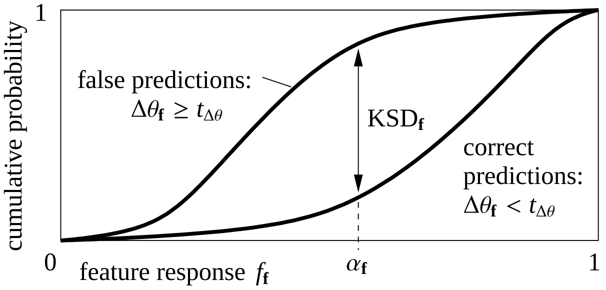

Figure 7. Kolmogorov-Smirnoff distance $\mathrm{KSD}_{\mathbf{f}}$ between the conditional distributions of feature response magnitudes given correct and false predictions.

is correct, and false otherwise. The optimal threshold $\alpha_{\mathbf{f}}$ is then given by a value that maximizes the Kolmogorov-Smirnoff distance $\mathrm{KSD}_{\mathbf{f}}$ between the two conditional distributions of $f_{\mathbf{f}}$ under the two conditions that the associated predictions are correct and false, respectively (Fig. 7). The feature model of $\mathbf{f}$ is then fitted to all data points θ with $f_{\mathbf{f}} > \alpha_{\mathbf{f}}$. The threshold $t_{\Delta\theta}$ is a global run-time parameter that can gradually be reduced over time to encourage the learning of improved features. A practical discussion of the Kolmogorov-Smirnoff distance is available elsewhere [20].

7 Feature Learning

In a manner similar to an exploring infant, our system learns on-line from experience by performing grasps and observing their outcomes, while associating their experience with object appearances. Specifically, our algorithm operates by proposing hand orientations based on feature observations, and observing the outcomes of grasping procedures (cf. Fig. 3). When a recommendation was poor, new features are added to the repertoire in parallel to the execution of grasping tasks. All features are evaluated over the course of many future grasps. Each feature is annotated with its parameters μ_k, κ_k, $P(k)$, and with the total number N of valid data points modeled by this mixture. Moreover, for each feature, the system maintains an estimate of the expected grasp quality $\hat{E}[\mu_0]$ associated with recommendations derived from it. Two separate feature sets, $\mathcal{F}_2$ and $\mathcal{F}_3$, are maintained for two- and three-fingered grasps. Initially, both feature sets are empty.

To allow easy recomputation of statistics of grasp experiences, the system maintains an *instance list*. Each item in the list contains all important data pertaining to an experience, including the response magnitudes and orientations of each observed feature, their prediction errors $\Delta\theta_{\mathbf{f}}$, as well as the actually achieved grasp quality μ_0 and the hand orientation θ_H used. The instance list can be cut off after some number of experiences, keeping its size constant. Since the feature set evolves – new features are added, old features are obsoleted – only little, if any, relevant information will be lost.

On receipt of a new object, an overhead image of it is searched for all features $\mathbf{f}_c \in \mathcal{F}_c$, $c \in \{2, 3\}$. The strongest occurrence of each feature with $f_{\mathbf{f}_c} > \alpha_{\mathbf{f}_c}$ are retained. By rejecting those features that are present with a value inferior to their specific optimal threshold $\alpha_{\mathbf{f}_c}$, those features are ruled out that were learned for other objects and are thus not predictive for the current one.

Of the retained features, the one that maximizes the expected grasp quality (i.e., that minimizes the expectation of μ_0) is selected to form a recommendation of grasp parameters. The number of fingers to be used is given by c, since $\mathbf{f}_c \in \mathcal{F}_c$ was learned for c-fingered grasps. If the angular mixture model associated with $\mathbf{f}_c$ has $K > 1$ modes, then the narrowest mode k is selected (i.e., maximal κ_k) as this will provide the most precise predicted hand orientation. This orientation is given by $\theta_h = \theta_{\mathbf{f}_c} + \mu_k$, where $\theta_{\mathbf{f}_c}$ is the orientation of $\mathbf{f}_c$ measured in the current image. The grasp is then performed using c fingers and an initial hand orientation of θ_h.

For learning purposes, predictions based on all other features are also computed, though not executed. The data characterizing the present experience are added as a new entry to the instance list. Then, based on the updated statistics gathered from the instance list, all feature models are updated by recomputing the $\mathrm{KSD}_{\mathbf{f}}$ and re-estimating their mixture models based on their updated thresholds $\alpha_{\mathbf{f}}$.

If the prediction derived from $\mathbf{f}_c$ was inaccurate, this indicates that the current feature set is insufficient for dealing with the present situation. In this case, two new features are added to $\mathcal{F}_c$: One is created by sampling a random texel or a random pair of edgels from the current image; the other is generated by expanding a randomly chosen existing feature $\mathbf{f} \in \mathcal{F}_c$ geometrically by adding a randomly chosen nearby edgel or texel. The latter method allows the emergence of increasingly descriptive features, which is essential for learning object-specific grasp recommendations. Over the course of many future grasps, feature models are formed for these new features. If a model turns out to be highly predictive in terms of a large $\mathrm{KSD}_{\mathbf{f}}$, then the associated feature will be used successfully for grasp recommendations; otherwise, it will rarely or never be used and will eventually be removed from the set.

To increase the chance that sampled features correspond to meaningful image structures, the sample space concentrates on *salient* image locations. The choice of the saliency criterion is not critical, as long as it admits a sufficiently large number of image locations. It is important to note that this restriction to salient points is not a necessary part of the learning algorithm, as the learning procedure discovers automatically, with experience, whether a feature is useful or not. It simply accelerates the learning process by focusing the search on image points that are, in an intuitive sense, likely to be meaningful.

8 Experimental Results

The learning algorithm just presented is designed to operate on-line. However, due to the relatively long execution times of Coelho's grasping system on the physical robot, a thorough on-line evaluation was impractical. Instead, we evaluated our system using a detailed kinematic and dynamic simulator of our hand/arm system (Fig. 1). Grasped objects of three categories were generated synthetically using probability distributions over size and angular parameters. For the purpose of visual feature learning, noisy photo-realistic renderings were produced (see Fig. 10 below). The visual system was trained off-line using the results of dozens of simulated grasps using two or three fingers on each of the three object categories. No features were ever learned for the cylindrical object category, as such objects are well grasped using either number of fingers at any hand orientation.

Two experiments were performed to evaluate the effect of visual priming on the haptic grasping procedure. The two primary evaluation criteria were the number of haptic probes (fingertip displacements) executed before the grasp controller converged, and the quality of the achieved grasp on convergence in terms of the friction coefficient μ_0.

In the first experiment, a two-fingered grasp controller was cued using features trained off-line, iteratively using the best 20 two-fingered grasps of cubes available in the training set. An analogous experiment was performed using three-fingered grasps of triangular prisms (Fig. 8). For cubes, the number of lengthy grasps that required more than about 13 haptic probes was drastically reduced. Such grasps typically do not converge to a stable configuration. Likewise, the number of extremely fast grasps that required only a single probe increased substantially. For both cubes and triangular prisms, the expected number of probes was somewhat reduced, and the number of poor grasps (with large μ_0) dropped dramatically.

In a second experiment, a set of visual feature models was trained using a training set of 80 grasps, including the best 20 grasps each of cubes and triangular prisms using two- and three-fingered grasp controllers. Both the two- and three-fingered visual feature models were exposed both to cubes and to triangular prisms, and the learning procedure learned features that were specific to cubes or triangular prisms exclusively. Thus, these features gathered meaningful statistics of the experienced friction coefficients, which could then be used to recommend a two- or a three-fingered grasp.

The results are shown in Fig. 9. The first two columns display the results achieved by the purely haptic system, for two- and three-contact controllers. In both cases, grasped objects included both cubes and triangular prisms. This explains the bimodal distributions of μ_0 shown in the bottom row: The modes centered near $\mu_0 = 0.2$ mostly correspond to two-fingered grasps of cubes and three-fingered grasps of triangular prisms, while the modes centered near $\mu_0 = 0.6$ tend to correspond to three-fingered grasps of cubes and two-fingered grasps of triangular prisms. This illustrates that neither two- nor three-fingered controllers alone are sufficient to execute high-quality grasps reliably for both cubes and triangular prisms.

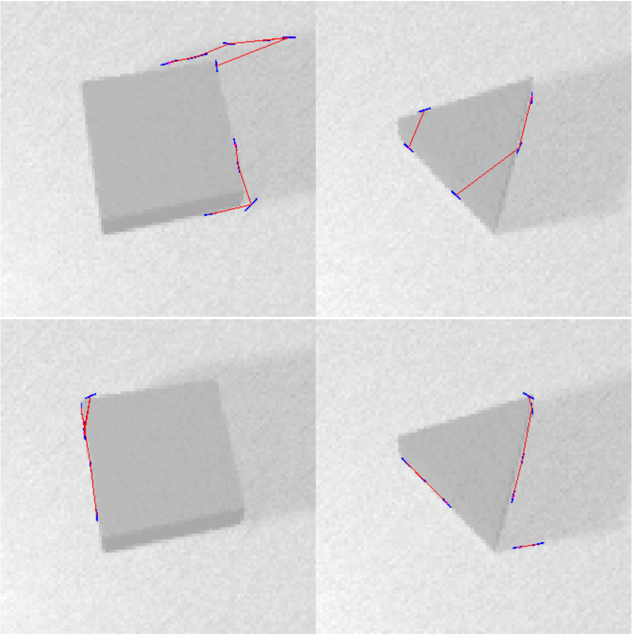

Figure 10. Examples of learned features.

The right column shows the results achieved if the learned visual models determine which grasp controller to use, and how to orient the hand at the onset of a grasp. Almost all grasps result in a very good μ_0; the second mode has almost completely disappeared. Moreover, very long trials (more than about 15 probes) are practically eliminated (cf. the two-fingered native controller in the left column).

Figure 10 shows examples of the features learned. Interestingly, one feature captures both the cube and the boundary of its shadow. Apparently there were enough fitting examples in the data that such a feature was advantageous. Another feature rests entirely on the boundary of the shadow. This may well be one whose lack of utility was yet to be discovered by the learning algorithm.

9 Conclusions

We described a visual learning system that serves to preshape a dextrous robotic hand prior to grasping. Grasp pa-

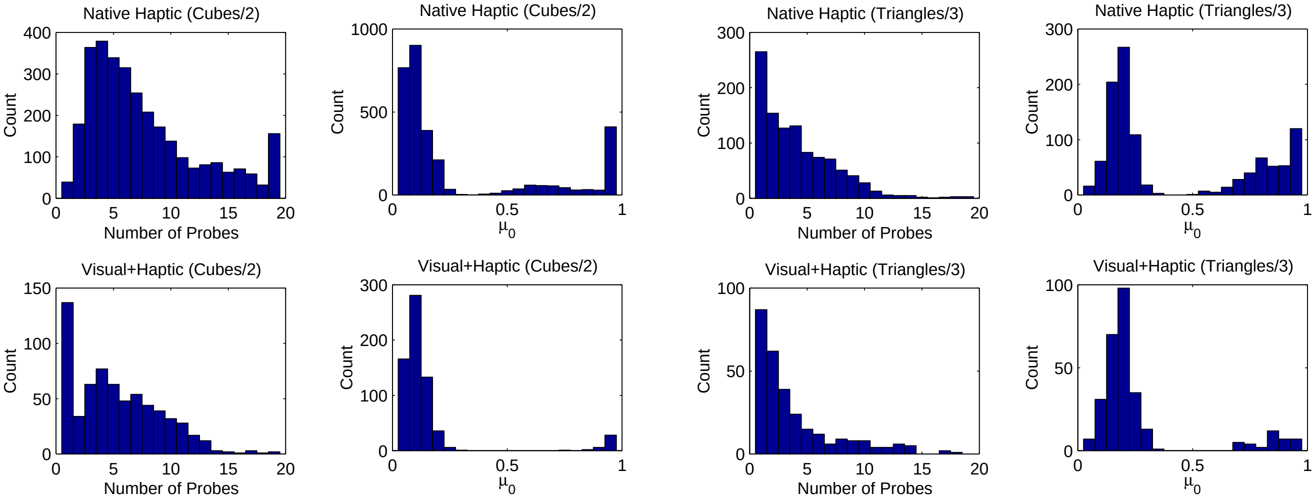

Figure 8. Results on two-fingered grasps of cubes (left), and three-fingered grasps of triangular prisms (right). The upper row shows purely haptic grasps; in the lower row, grasps were cued using learned visual features. The rightmost bin in each histogram includes all instances with a number of probes ≥ 20, or $\mu_0 \geq 1$, respectively.

rameters are derived from the occurrence and orientation of learned, individually highly predictive features. The computational demands of learning are negligible compared to the execution times of a physical robot. Thus, the performance of the trained system is practically instantaneous. The number of grasps required for training is mainly determined by the ease of finding pertinent features. For the above examples, a few hundreds of grasps were used to train the system. This constitutes a practical limitation of the present system. Another limitation is the fact that it learns only a 2-D orientation parameter. One way to extend it to three dimensions would be to use feature representations that permit the determination of affine pose parameters. This is an area of future research.

The trained system consistently produces grasps superior to the blind system. The learning procedure resembles infant exploratory behavior in that haptic grasping experiences are associated with visual observations, that are later used to form predictions for the haptic system to improve grasping efficiency. To enhance the robustness of the system, future research will investigate methods for learning features that are highly predictive as populations.

The combined grasping system resembles human grasping of familiar objects in several respects: The hand is pre-shaped prior to tactile contact using appearance cues learned from experience, the grasp itself relies on haptic feedback, and no explicit object recognition, scene reconstruction, or kinematic planning are performed. On biological grounds, we believe that a key to building successful vision-guided interactive systems (such as household robots and autonomous vehicles) lies in methods for controlling actuators directly on the basis of relatively low-level visual cues [1]. This will generally require on-line learning, and will often benefit from integration with non-visual sensory modalities.

References

[1] D. H. Ballard and C. M. Brown. Principles of animate vision. *Computer Vision, Graphics, and Image Processing: Image Understanding*, 56(1):3–21, July 1992.

[2] H. B. Barlow. Summation and inhibition in the frog's retina. *J. Physiol. (Lond.)*, 119:69–88, 1953.

[3] C. Biernacki, G. Celeux, and G. Govaert. Assessing a mixture model for clustering with the integrated completed likelihood. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(7):719–725, July 2000.

[4] J. A. Coelho, Jr. and R. A. Grupen. A control basis for learning multifingered grasps. *Journal of Robotic Systems*, 14(7):545–557, 1997.

[5] J. A. Coelho, Jr. and R. A. Grupen. Learning in nonstationary conditions: A control theoretic approach. In *Proc. 17th Int. Conf. of Machine Learning*. Morgan Kaufmann, 2000.

[6] N. I. Fisher. *Statistical Analysis of Circular Data*. Cambridge University Press, 1993.

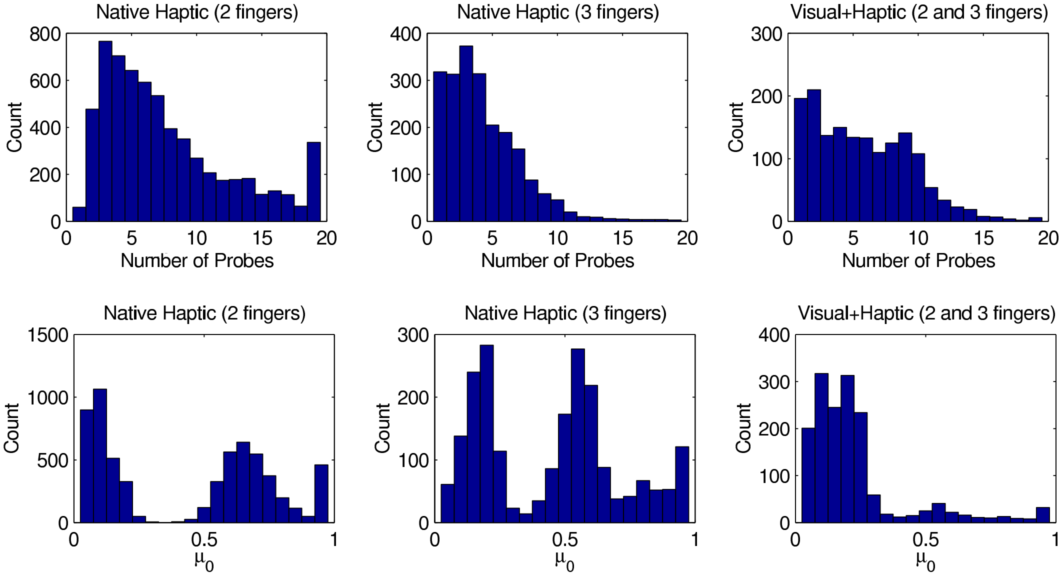

Figure 9. Grasp results on cubes and triangular prisms. The number of grasp contacts was chosen using the learned visual feature models. The left two columns show purely haptic grasps; in the right column, grasps were cued using learned visual feature models. The rightmost bin in each histogram includes all instances with a number of probes ≥ 20, or $\mu_0 \geq 1$, respectively.

[7] W. T. Freeman and E. H. Adelson. The design and use of steerable filters. *IEEE Trans. Pattern Anal. Mach. Intell.*, 13(9):891–906, 1991.

[8] D. Ingle. Two visual systems in the frog. *Science*, 181:1053–1055, 1973.

[9] M. Jägersand and R. Nelson. Visual space task specification, planning and control. In *Proc. IEEE Symp. on Computer Vision*, pages 521–526, 1995.

[10] J. J. Koenderink and A. J. van Doorn. Representation of local geometry in the visual system. *Biological Cybernetics*, 55:367–375, 1987.

[11] D. Marr. *Vision*. Freeman, San Francisco, 1982.

[12] A. D. Milner and M. A. Goodale. *The Visual Brain in Action*. Oxford University Press, 1995.

[13] S. K. Nayar, H. Murase, and S. A. Nene. Learning, positioning, and tracking visual appearance. In *Proc. Int. Conf. on Robotics and Automation*, San Diego, CA, May 1994. IEEE.

[14] J. H. Piater and R. A. Grupen. Toward learning visual discrimination strategies. In *Proc. Computer Vision and Pattern Recognition*, volume 1, pages 410–415. IEEE Computer Society, June 1999.

[15] J. H. Piater and R. A. Grupen. Feature learning for recognition with Bayesian networks. In *Proc. Int. Conf. on Pattern Recognition*, Sept. 2000.

[16] R. P. N. Rao and D. H. Ballard. An active vision architecture based on iconic representations. *Artificial Intelligence*, 78:461–505, 1995.

[17] W. E. Reichardt and T. Poggio. Visual control of flight in flies. In W. E. Reichardt, V. B. Mountcastle, and T. Poggio, editors, *Recent Theoretical Developments in Neurobiology*. 1979.

[18] C. Schmid. A structured probabilistic model for recognition. In *Proc. Computer Vision and Pattern Recognition*, volume 2, pages 485–490, June 1999.

[19] G. Schwarz. Estimating the dimensions of a model. *Annals of Statistics*, 6:461–464, 1978.

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

[21] R. A. Young. The Gaussian derivative model for spatial vision: I. Retinal mechanisms. *Spatial Vision*, 2(4):273–293, 1987.