Iterative Learning of Grasp Adaptation through Human Corrections

Eric L. Sauser, Brenna D. Argall, *Member, IEEE*, Giorgio Metta, and Aude G. Billard, *Member, IEEE*

***Abstract*—In the context of object interaction and manipulation, one characteristic of a robust grasp is its ability to *comply with external perturbations* applied to the grasped object *while still maintaining the grasp*. In this work we introduce an approach for grasp adaptation which learns a statistical model to adapt hand posture solely based on the perceived contact between the object and fingers. Using a multi-step learning procedure, the model dataset is built by first *demonstrating* an initial hand posture, which is then physically *corrected* by a human teacher pressing on the fingertips, exploiting compliance in the robot hand. The learner then *replays* the resulting sequence of hand postures, to generate a dataset of posture-contact pairs that are not influenced by the touch of the teacher. A key feature of this work is that the learned model may be further *refined* by repeating the correction-replay steps. Alternatively, the model may be *reused* in the development of new models, characterized by the contact signatures of a different object. Our approach is empirically validated on the *iCub* robot. We demonstrate grasp adaptation in response to changes in contact, and show successful model reuse and improved adaptation with additional rounds of model refinement.**

***Index Terms*—Force and Tactile Sensing, Learning and Adaptive Systems, Dexterous Manipulation, Multifingered Hands**

I. Introduction

Object interaction and manipulation is a challenging topic within robotics research. When a detailed model of the object shape and surface properties is known, one can reason about grasp optimality. However, the prior knowledge requirement is extensive - object properties like the mass distribution or surface texture can be difficult to obtain, for example requiring force sensors or accurate tactile sensing - and how these properties change as the object is manipulated can be difficult to predict. When detailed information about the object shape and surface properties is not known, compromises like grasp sub-optimality and a strong reliance on accurate runtime sensing must be made. Object manipulation becomes even more challenging within the context of dynamic interactions, when the grasp on the object is not static.

In this work, the target behavior is *grasp adaptation*; that is, the ability to be intentionally responsive to external forces so as to comply smoothly with external perturbations, all while maintaining contact with the object (Fig. 1a). The use of force or impedance feedback controllers offer robust solutions to the goal of maintaining contact with an object; however, most works do not consider the additional goal of being intentionally *compliant* and to follow perturbations [1], [2], [3], [4]. Smooth compliance in response to object perturbations when grasping necessitates a tight coordination between all fingers, else the grasped object might fall from the hand. Moreover, this coordination is typically ensured by a good knowledge of the hand kinematics and of the object shape [5], [6], [7], [8]. To tackle this issue, rather than handcraft the coordination patterns across all fingers for each novel object, we adopt a learning approach based on human demonstration. The coordination patterns thus are extracted from a set of good example grasps. The use of demonstration learning is motivated further by the high-dimensionality of the task state-space, due to the number

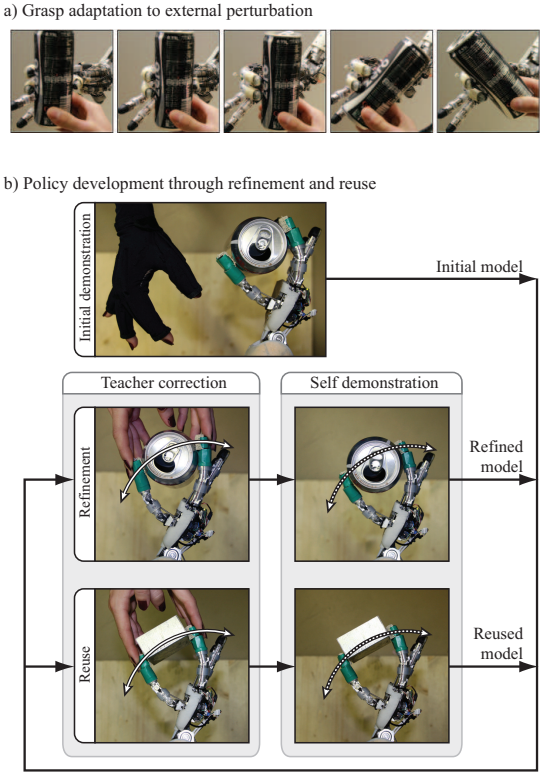

Fig. 1. a) *Grasp adaptation:* When an external perturbation is applied on the object currently grasped by the robot, the robot dynamically adapts its grasp to comply with the perturbation. b) *Overview of our approach for learning grasp adaptation skills:* An adaptive model for maintaining a grasp in response to changing contacts is built and updated (top→bottom) by having a teacher demonstrate a grasp and then refine the range of possible grasps for adaptation through corrective feedback (left column). Robot self-demonstration (right column) is necessary for acquiring sensory information that is not influenced by the touch of the teacher. Furthermore, the development of a new model that is responsive to a new object is also possible through model reuse.

E. Sauser, B. Argall and A. Billard are with the Learning Algorithms and Systems Laboratory (LASA) at the Ecole Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland email: [eric.sauser,brennadee.argall,aude.billard]@epfl.ch. G. Metta is affilitated with the Italian Institute of Technology and the LIRA-Lab at the University of Genoa, Genoa, Italy email: giorgio.metta@iit.it.

of degrees of freedom in the fingers and the sensory signals at play. Showing by example can simplify the specification of coordinated postures between all of the fingers. If the examples are shown kinesthetically, by physically touching the robot to move its fingers, demonstration also allows the teacher to provide the robot with an intuitive notion of force.

Our work takes the approach of learning a statistical model able to predict a desired hand posture and fingertip pressure from the current signature of the contact perceived at the robot's fingertips. The approach depends on tactile sensing at the fingertips and human demonstration to provide an example set of feasible grasps.[1] The approach does not require any kinematic nor dynamic model of the hand nor object, unlike model-based manipulation approaches. Such requirements of a detailed model and consequently, precise sensing capabilities, in practice can be an issue for many robotic platforms. Instead, the use of a probabilistic model allows for the encapsulation of the intrinsic non-linear mapping between the noisy tactile data and joint information, obtained directly from example grasps.

The dataset of examples is built both from human demonstration, and from self-demonstration by the robot after correction by a human teacher. In particular, our model derives from a multi-step learning procedure, that iteratively builds a training dataset from a combination of teacher *demonstration*, teacher *correction* and learner *replay* (Fig. 1b). Corrections are accomplished by having the teacher directly act on the fingers of the robot. In contrast to other demonstration mechanisms like vision systems or data gloves, we suggest that directly acting on the fingers allows the human to detect the forces applied to the grasped object, and thus to achieve a better demonstration of the applied forces. The dataset also is built *iteratively*, as the teacher interactively corrects the robot's executions and thus refines the learned behavior. A key distinction in our work when compared to other iterative demonstration learning approaches [9], [10], [11], [12], [13] is the focus on perturbations, that possibly take the execution far from what has been shown in the demonstration set. Our novel formulation for avoiding over-generalization also ensures that the robot's response is always valid with respect to the example dataset. Our corrections furthermore aim not only to improve upon a demonstrated behavior, but also to explicitly show additional flexibility and adaptation beyond an executed pose.

Our approach is empirically validated on the *iCub* robot [14], building contact models for multiple objects of different shapes and sizes. The effectiveness of the iterative learning procedure is confirmed, by measuring an increase across models in the joint ranges encompassed by a given model, as well as in the smoothness of the adaptation and the fingers' ability to maintain contact with the object when faced with perturbations. Although we overlook the analytical force-closure constraint [15] during model training, we show that the grasps learned using our approach do in fact satisfy the constraint of force-closure. The benefit of self-replay following teacher correction furthermore is demonstrated.

The following section provides an overview of the related literature that supports and motivates this work. Section III then formally introduces our approach to iteratively learn an adaptation model, along with the details of the control method for grasp adaptation. Hardware specifications and the experimental setup are detailed in Section IV, and results on the *iCub* humanoid in Section V. Section VI concludes with a summary and discussion of contributions, and directions for future work.

[1]We assume the training dataset consists of only valid grasps, such that the grasped object doesn't slip or fall from the hand, as ensured by the teacher's supervision.

II. Related Work

This section provides an overview of related literature within the topics of dexterous manipulation, tactile sensing technologies, reactive grasping and demonstration learning.

A. Dexterous Manipulation

In dexterous manipulation, one important task is to determine the required actuator forces/torques to maintain a grasp to an object [16], [6], [17]. When maintaining grasp contact while modifying the current posture, grasp stability is an important issue [15]. Indeed, if pose transitions are not managed carefully, undesirable behavior can appear, e.g. the object can fall from the hand. This control problem is hard, especially given that a robotic hand is usually composed of a high number of degrees of freedom, and that precise tactile sensing is difficult to obtain, and also inherently sensitive to motion and sensor signal discontinuities [18].

In order to cope with these issues, model-based approaches were developed, which are based on known kinematic and dynamic properties of the hand and object [6], [19]. However, these approaches require a quasi-perfect knowledge of the geometric relationships within the dexterous manipulator-object system. A high quality model of the hand is thus necessary to achieve very precise manipulator control and sensing, and such a model is not available for many robot hands [18], [1]. Moreover, because of the high complexity of the problem, motion and manipulation are usually preplanned, during which the quality of the grasp or intended manipulation is also optimized through various techniques and criteria [17], [19], [20]. Therefore, the application of these methods is reduced in general to constrained and controlled environments, and rarely adapts online in realtime. Another drawback is that such approaches typically require high specialization with respect to specific hand-object combinations, and thus struggle with the challenge of generalization to novel objects. With respect to this issue, our incremental learning procedure suggests to reuse an existing model to bootstrap the development of a new model for a similar object. We will show that this procedure can efficiently reduce the time required to develop new models.

Furthermore, recent work has shown the necessity of having access to a rich set of sensory information in order to perform manipulation tasks of increasing complexity [21], [22], [5], [23]. Through the use of such advanced sensing devices, touch-based exploratory methods have developed that discover and learn object properties and manipulation strategies [19], [24], [25]. In our work, tactile feedback and control similarly are learned from experience that the robot acquires by itself through manipulation. However, we also take advantage of the

teacher's expertise within a programming by demonstration framework in order to constrain the exploration to areas of the sensory space that contain valid grasp only.

B. Reactive Grasping and Contact Maintenance

A common motivation for reactive grasping strategies is to circumvent the need for a detailed object model. By means of low-level reflexes [26] or high-level behavioral rules [4], a grasp to an improperly modeled object can still be achieved. Along this line, more complex methods incorporate sensory data to improve the current representation of the environment. The limitation here therefore is the requirement of precise sensing, which is not available for many robot platforms. Data gathered through reactive grasping strategies is used to estimate the position and orientation of a novel object [27], to systematically gather information about the object shape [28], [29] and to infer areas where the fingertip might safely be moved to gather more sensor data [30]. Other approaches go further and gather shape information with the intent to build an explicit object model [31].

Another practical application for reactive grasping strategies is to maintain contact after a grasp is established. The continued development of sensor technologies with increasing sophistication [21], [22], [5], [23] promoted the use of adaptive control schemes such as force and impedance control [16], [7], and later hybrid methods [5]. These approaches are hierarchically combined with high-level and predefined behavioral laws that triggers the controllers when appropriate. For example, early work proposed idea of using reflexes to refine and maintain the grasp [26]. Security reflexes are employed to recover a loss of contact [32], and fuzzy logic rules specify a change in contact normal based on perceived forces [33]. Contact recovery behaviors are triggered by tracking temporal tactile data to detect slip [5], [34]. A grasping force is applied to counteract the perturbing force that results from object manipulation by the robot [8], and force feedback control is used for stabilizing the grasps during explicit finger repositioning for object rotation and translation behaviors [3]. Other approaches pair upper-level controllers that target grasp points with lower-level reactive controllers that avoid collisions [30], [2].

As mentioned in the introduction, our work is distinguished from existing reactive grasping approaches by its aim to be compliant to external perturbations; furthermore, this compliance is learned rather than being hand-coded. The learned statistical model determines how to coordinate the motion of all the fingers when responding to external forces. Our grasp adaptation paradigm however does employ hierarchical control techniques similar to those used in reactive grasping, though with a novel formulation for smoothly switching from higher priority position control to force control. The switch occurs if the current pose is sufficiently close - according to a metric learned by the probabilistic model - to the target pose.

C. Robot Learning

Information gathered through reactive grasping procedures also is used within learning contexts. Data gathered through interactive trial and error is used to grasp arbitrary objects [28], and grasp quality is improved by learning better grasp point locations [2] or responding to pose estimates from a learned probabilistic model [30].

Another promising research direction for helping to reduce the complexity of dexterous manipulation is demonstration-based learning approaches [34], [2], [35], [36], [37], [38]. All of these methods share the intuitive advantage of being relatively simple for a human user to transfer task knowledge to a robot. Within this line of supervised manipulation, the robot is directly taught by a human user how to achieve a grasp [37], [38] through a variety of human-robot interfaces such as complex computer vision systems [36], [37] or data gloves [39]. These data capture methods however do not allow a human to perceive the forces that the robot actually applies to the grasped object.

Human demonstrations are not used for grasp pose description only. For example, in the context of grasp planning, demonstration data has been treated as statistical priors to reduce the computational cost for searching for optimal solutions [35]. Another example is to use the variability between demonstrations to teach a robot in what way, and by how much, to react and adapt to environmental perturbations [37].

Learning procedures lend themselves naturally to iterative dataset building and behavior refinement. For example, a human teacher might supervise the learning process, by modifying targets learned from demonstration [9] or resolving ambiguities in goal representations [10]. Datasets are iteratively built by providing new demonstrations in areas of low policy prediction confidence [40], [41], by providing explicit corrections on policy predictions to generate new data [40], [12] and by physically touching a robot during execution to provide kinesthetic corrections [11], [42], [13].

As discussed in the introduction, a key distinction between our work an other policy refinement approaches is the focus on perturbations, which possible take the learner far from what was demonstrated, and the intention to show flexibility and adaptation beyond what was demonstrated. Furthermore, our executions do not depend on time (unlike [11], [42], [13]), as our goal is not to execute a trajectory but rather to respond online to changes in contact with an object.

III. Approach and Methodologies

We now overview the details of our approach for iterative grasp adaptation learning. In this work, we consider the case where the location of the contacts between each finger and the object remains fixed throughout adaptation (Fig. 1a). First, we will describe the variables at play in our approach, as well as the architecture of our system (Sec. III-A). We will then describe our algorithm for iteratively building the adaptation model by generating a training dataset over multiple steps under teacher supervision (Sec. III-B). Further, we provide a technical description of the statistical model (Sec. III-C) and its use during behavior execution (Sec. III-D).

A. System Architecture

The state of our system is described by three main variables. The *contact signature* $\boldsymbol{\phi} \in \mathbb{R}^{N_\phi}$ corresponds to the

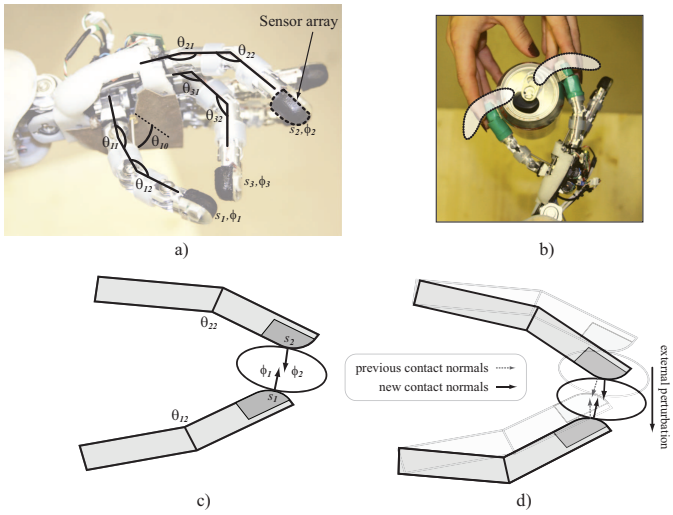

Fig. 2. *a)* The *iCub* hand and corresponding joint angles of the 3 digits used in our empirical validations. Each black fingertip of the hand consists of a tactile sensor array. *b)* By pressing on the fingertips, the teacher demonstrates to the robot the range of candidate hand poses for adaptation (light surfaces). *c)* In this work, the contact signature $\boldsymbol{\phi}$ of each finger corresponds to the contact normal. *d)* A perturbation applied on the object results in a change in contact signature. With this information, our learned reactive controller is able to adapt the grasp in order to maintain the contact with the object.

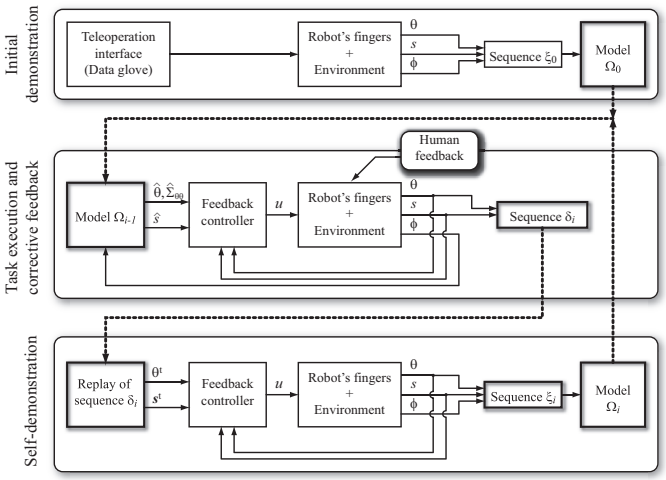

Fig. 3. Schematic overview of our system and approach. The top panel corresponds to the initial *demonstration*, where the robot's hand is controlled by the human teacher through teleoperation; the middle panel to model *refinement* or *reuse* though corrective feedback while the robot is executing its current model; the bottom panel to *self-demonstration* of the sequence of corrected poses in order to obtain a training dataset that is not corrupted by the touch of the teacher. From these data, a new model is learned, which may further be refined or reused.

3-dimensional contact normal direction at each of the N_f fingertips when in contact with an object, composed into a single vector ($N_\phi = 3 \cdot N_f$).[2] The *hand pose* $\boldsymbol{\theta} \in \mathbb{R}^{N_\theta}$ denotes the joint configuration of a robotic hand having N_θ degrees of freedom (DoF). Finally, the *contact pressure* $\boldsymbol{s} \in \mathbb{R}^{N_s}$ corresponds to the pressure values measured on each fingertip.[3] An illustration of these variables is shown in Figure 2.

In our approach, we assume that these three variables are sufficient to determine the grasp of an object. As illustrated in Figure 1b, we iteratively gather datasets of such grasp variables from teacher demonstration, and subsequently, from teacher correction. We then learn an estimate of the joint distribution of these three variables as a probabilistic model Ω. During behavior execution, the model is used to generate a mapping $\boldsymbol{\phi} \mapsto (\hat{\boldsymbol{\theta}}, \hat{\boldsymbol{s}})$ that predicts a target hand pose and desired contact pressure given the current signature of the contact between each fingertip and the object.[4] As shown in Figure 3(middle), which provides a schematic overview of our system, these predictions are then fed as control signals to a feedback controller that generates torque commands to the finger motors.

B. Iterative Building of the Dataset

We now provide the details of iteratively building the prediction model, by generating a dataset over multiple steps through teacher supervision. This method is summarized in Algorithm 1.

[2]Taken more generally, the contact signature could refer to a variety of metrics (e.g. tangential force vector, contact area), depending on the task and robot platform.

[3]In our implementation, we sum the pressure measurements of each sensor located on the same fingertip to a single value per finger, and so $N_s = N_f$.

[4]We adopt the notation $\hat{x}$ for a target value of prediction variable x.

Algorithm 1

Given a model Ω_{i-1}

1. Tactile Correction
for $t \in \{1..T\}$
Model Ω_{i-1} predicts $(\hat{\boldsymbol{\theta}}^t, \hat{\boldsymbol{s}}^t)$ given current $\boldsymbol{\phi}^t$.
Controller executes target $(\hat{\boldsymbol{\theta}}^t, \hat{\boldsymbol{s}}^t)$
Teacher adjust pose, resulting in a measurement $(\boldsymbol{\theta}^t, \boldsymbol{s}^t)$.
end
Result: Sequence $\boldsymbol{\vartheta}_i = \{(\boldsymbol{\theta}^t, \boldsymbol{s}^t)\}_{t=1}^{T}$ of pose-pressure pairs.

2. Self-Demonstration
Controller executes target sequence $\boldsymbol{\vartheta}_i$.
Result: Sequence $\boldsymbol{\xi}_i = \{(\boldsymbol{\theta}^t, \boldsymbol{s}^t, \boldsymbol{\phi}^t)\}_{t=1}^{T}$ of pose-pressure-contact tuples.

3. Model Learning
Use data $\boldsymbol{\xi}_i$ to train a new prediction model Ω_i.

1) Demonstration: In the absence of an existing model, an initial target hand pose is demonstrated by the teacher (Fig. 1b). A small dataset of pose-pressure-contact tuples $\boldsymbol{\xi}_0 = \{(\boldsymbol{\theta}^t, \boldsymbol{s}^t, \boldsymbol{\phi}^t)\}_{t=1}^{T}$ are recorded (see Fig. 3(top)). From these data, an initial task model Ω_0 is learned (Sec. III-C). As model development is done along several iterations, we will index each variable accordingly. For instance, a model learned after the i^{th} iteration will be denoted by Ω_i.

2) Tactile Correction: In our approach, a model of the task can be refined multiple times. During the i^{th} iteration, the teacher provides corrective feedback while the robot executes the task using the previously learned prediction model Ω_{i-1}. Given the current contact signature $\boldsymbol{\phi}$, the model sends control signals $(\hat{\boldsymbol{\theta}}, \hat{\boldsymbol{s}})$ to the hand controller $\boldsymbol{u}$ (Fig. 3(middle)). Concurrently, the teacher provides corrective feedback directly

on the robot's fingers.[5] Figure 2b provides an illustration of tactile correction under our implementation, where the teacher gently pulls or presses on the robot fingers to reposition them within their compliance limits. As during tactile correction, the teacher changes the hand posture and accordingly also the contact signature $\boldsymbol{\phi}$, the model predicts new targets $(\hat{\boldsymbol{\theta}}, \hat{\boldsymbol{s}})$ for the controller. The result is a sequence $\boldsymbol{\vartheta}_i = \{(\boldsymbol{\theta}^t, \boldsymbol{s}^t)\}_{t=1}^{T}$ of T pose-pressure pairs. Contact signature $\boldsymbol{\phi}$ is not recorded, since it is considered to be unreliable on account of the contact with the teacher's hand in addition to the object.[6]

The teacher provides corrections to (i) encourage better contact with the object and (ii) shift the pose as much as possible within the compliance constraints of the hand, while still maintaining contact. Whether the corrections are intended to *refine* the current model or develop a new model depends on whether the hand is interacting with a novel object. If yes, then a new model is being developed from the *reuse* of the current model. Note that for reuse to be feasible (i) the novel object must be of a size that is within the compliance limits of the robot manipulator when maintaining a posture appropriate for grasping the original object, and (ii) the set of admissible hand pose for the novel object should partly overlap that of the original object.

3) Self-Demonstration: This phase generates the data which will be actually used to train the new prediction model. As shown in Figure 3(bottom), the sequence $\boldsymbol{\vartheta}_i$ of pose-pressure pairs from the tactile correction phase are sequentially fed as targets to the feedback controller. During this phase, it is the role of the teacher to verify that the execution of this control sequence produces a set of valid grasp, i.e., ensuring that the object doesn't fall from the hand. If not, the correction phase is restarted. As a result, a sequence $\boldsymbol{\xi}_i = \{(\boldsymbol{\theta}^t, \boldsymbol{s}^t, \boldsymbol{\phi}^t)\}_{t=1}^{T}$ of T pose-pressure-contact tuples is obtained. Note that now, in the absence of any touch from the teacher, all of the variables relating to object contact ($\boldsymbol{s}$,$\boldsymbol{\phi}$) are considered to be *reliable* and thus are recorded.

C. Model Learning

The final phase of the algorithm is to learn a statistical model Ω_i from the recorded data $\boldsymbol{\xi}_i$. Without loss of generality, we omit the index i in the rest of this section to lighten the notation.

1) Statistical Model: We model the self-demonstration data as a *Gaussian Mixture Model* (GMM) [43], and hence get a probabilistic encoding of the joint distribution of the variables, i.e., $p(\boldsymbol{\theta}, \boldsymbol{s}, \boldsymbol{\phi} \,|\, \Omega)$. This choice of probabilistic encoding has the advantages of capturing the non-linear correlations present in the demonstration and sensor data, as well as of encapsulating the inherent noise present the sensory signals. Furthermore, *Gaussian Mixture Regression* (GMR) provides a closed-form solution to compute the conditional $p(\boldsymbol{\theta}, \boldsymbol{s} \,|\, \boldsymbol{\phi}, \Omega)$ of a GMM [43], which allows us to predict a desired finger posture $\hat{\boldsymbol{\theta}}$ and contact sensor reading $\hat{\boldsymbol{s}}$ given the current contact signature $\boldsymbol{\phi}$. The ability of GMM/GMR to generalize and extrapolate well over missing and unseen data has been shown to be efficient in many experimental settings [44], [45]. In unseen contexts, other non-linear regression methods such as Gaussian Process Regression converge to a default mean value. With respect to our task, if this value is badly tuned, it can lead to unstable grasps, and therefore to task failure. The probabilistic encoding of GMM/GMR also has the advantage of being able to determine whether a point in the input space is likely under the learned model. This ability to determine the likelihood of a query point is a crucial property, as outside the regions covered by the training data inference can be unreliable, and hence possibly poor. As will be described in Section III-D, we take advantage of this property to ensure the validity of our model prediction.

In a GMM, the joint probability distribution of all variables is encoded as a sum of K Gaussian components,

$$p(\boldsymbol{\theta}, \boldsymbol{s}, \boldsymbol{\phi} \,|\, \Omega) \;=\; \sum_{k=1}^{K} p_k \, p(\boldsymbol{\theta}, \boldsymbol{s}, \boldsymbol{\phi} \,|\, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \tag{1}$$

where p_k is the prior of the k^{th} multidimensional Gaussian component and $\boldsymbol{\mu_k}, \boldsymbol{\Sigma_k}$ are respectively its mean and covariance, such that

$$\boldsymbol{\mu}_k \;=\; \begin{pmatrix} \boldsymbol{\mu}_{\boldsymbol{\theta},k} \\ \boldsymbol{\mu}_{\boldsymbol{s},k} \\ \boldsymbol{\mu}_{\boldsymbol{\phi},k} \end{pmatrix} \quad \boldsymbol{\Sigma}_k \;=\; \begin{pmatrix} \boldsymbol{\Sigma}_{\boldsymbol{\theta\theta},k} & \boldsymbol{\Sigma}_{\boldsymbol{\theta s},k} & \boldsymbol{\Sigma}_{\boldsymbol{\theta\phi},k} \\ \boldsymbol{\Sigma}_{\boldsymbol{s\theta},k} & \boldsymbol{\Sigma}_{\boldsymbol{ss},k} & \boldsymbol{\Sigma}_{\boldsymbol{s\phi},k} \\ \boldsymbol{\Sigma}_{\boldsymbol{\phi\theta},k} & \boldsymbol{\Sigma}_{\boldsymbol{\phi s},k} & \boldsymbol{\Sigma}_{\boldsymbol{\phi\phi},k} \end{pmatrix} \tag{2}$$

In order to train the model parameters from the data, the *Expectation-Maximization* algorithm is used [46]. In our experiments, our dataset contains between 1000 and 2000 datapoints, and the value of K is set using the *Bayesian Information Criterion* (BIC). Figure 4 shows an example probability density function estimated by a GMM on a self-demonstration dataset.

2) Modeling the Uncertainty of the Query Inputs: The model Ω is learned from a dataset containing only *valid* pose-pressure-contact tuples seen within the self-demonstration dataset. As this dataset was recorded in the absence of actual external perturbations, the demonstrated grasps belonging to this set have all fingers in contact with the object. In the presence of such perturbations, however, one or more fingers might lose contact with the object, producing a contact signature $\boldsymbol{\phi}$ that is random (due to sensor noise) and thus unreliable.

We therefore introduce a reliability measure α_j for each fingertip $j = \{1..N_f\}$. For the currently perceived contact signature $\boldsymbol{\phi_j} \in \mathbb{R}^3$ of each finger, its reliability measure $\alpha_j \in [0..1]$ is a value that depends on the current pressure reading s_j of the corresponding senor. We consider that the stronger the contact sensor reading, the more reliable the contact signature, and so

$$\alpha_j = \begin{cases} 0 & s_j \leq s^{\min} \\ (s_j - s^{\min})/(s^{\max} - s^{\min}) & s^{\min} < s_j < s^{\max} \\ 1 & s_j \geq s^{\max} \end{cases} \tag{3}$$

[5] We assume a robot manipulator that allows a human teacher to make small pose adjustments, either because of some inherent compliance, e.g. mechanical slack in the fingers, or the existence of explicit reactive motions.

[6] Arguably contact pressure $\boldsymbol{s}$, and not just contact signature $\boldsymbol{\phi}$, is also polluted by the tactile correction technique. However, empirically better performance was seen by using the controller with inputs $(\boldsymbol{\theta}, \boldsymbol{s})$ during the self-demonstration phase rather than just replaying the sequence of poses $\boldsymbol{\theta}$.

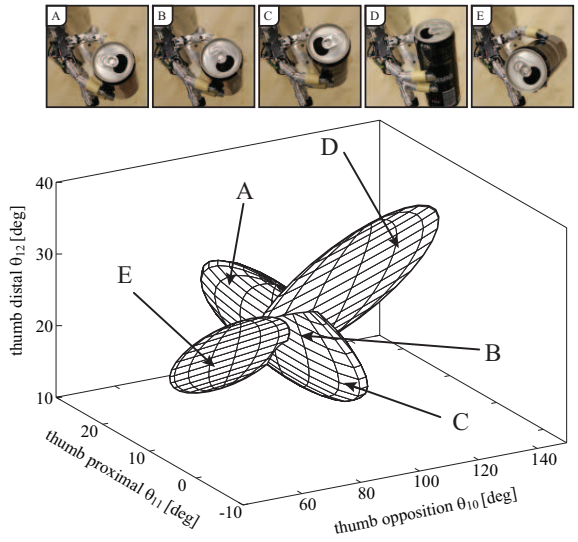

Fig. 4. Illustration of a GMM encoding within a subset of the joint angle dimensions (the three degrees of the thumb). Arrows indicate the corresponding location of example hand postures A-E (shown on top).

where $s^{\min}$ and $s^{\max}$ are threshold values on whether the contact signature is considered to be reliable or not.[7]

To incorporate this information, we derive a new joint probability distribution from the original learned model Ω, that additionally depends on the reliability $\boldsymbol{\alpha}$ of the input signal:

$$p(\boldsymbol{\theta}, \boldsymbol{s}, \boldsymbol{\phi} \,|\, \Omega, \boldsymbol{\alpha}) \quad = \quad \sum_{k=1}^{K} p_k \, p\big(\boldsymbol{\theta}, \boldsymbol{s}, \boldsymbol{\phi} | \boldsymbol{\mu}_k, \tilde{\boldsymbol{\Sigma}}_k(\boldsymbol{\alpha})\big) \quad (4)$$

where[8] $\boldsymbol{\alpha} = (\alpha_1, \alpha_1, \alpha_1, \ldots, \alpha_{N_f}, \alpha_{N_f}, \alpha_{N_f})^T$ and the new covariance matrices are given by

$$\tilde{\boldsymbol{\Sigma}}_{ij,k}(\boldsymbol{\alpha}) \quad = \quad \begin{cases} \boldsymbol{\Sigma}_{ij,k} + \mathrm{diag}(-\log(\boldsymbol{\alpha})), & \text{if } i = j = \phi \\ \boldsymbol{\Sigma}_{ij,k} & \text{otherwise} \end{cases}$$
$$\text{where } i, j \in \{\phi, \theta, s\}$$

The distribution thus now additionally considers a joint distribution from unreliable contact signatures to learned valid grasp configurations, the importance of which will become more apparent when describing the regression procedure in Section III-D. Note that $\boldsymbol{\alpha}$ is an additional prior on $p(\boldsymbol{\phi})$ given the current (potentially unreliable) sensor reading. It complements the variability learned by the model, which originally covers only the space of contact pressure readings seen during self-demonstration, all of which were the result of valid grasps.

D. Grasp Execution

Once a model is learned, it is used to predict the expected joint configuration $\hat{\boldsymbol{\theta}}$ and the expected pattern of pressure value $\hat{\boldsymbol{s}}$ at each fingertip, given the current contact signature $\boldsymbol{\phi}$. These two variables will then be used to generate the grasp by commanding the feedback controller, which will be described in Section III-D3.

[7]The value of these parameters is strongly sensor dependent: $s^{\min}$ should be set to a value slightly above the residual noise produced by the sensor when there is no contact, and $s^{\max}$ to a value corresponding to a decent pressure being applied to the fingertip sensor.

[8]Each α_i appears in triplicate to account for the 3 dimensions of $\phi_i \in \mathbb{R}^3$.

1) Projection to the Input Space: During the execution of the adaptation behavior, we first check whether the current query point $\boldsymbol{\phi}$ is likely enough with respect to the model[9]. If it is not, we use a projection $\boldsymbol{\phi}^\star$ of the query point $\boldsymbol{\phi}$ from which the model predicts the desired joint configuration and fingertip pressures. This projection is chosen such that $\boldsymbol{\phi}^\star$ is the closest point from $\boldsymbol{\phi}$ with a sufficient likelihood under the model. This operation is required for two major reasons. First, the prediction of a GMM in response to an input with low likelihood is a point that is likely to be far from the main distribution and hence, under our model, unlikely to be a valid grasp. Importantly, as generating an invalid grasp can have the consequence of leading to adaptation instability, we have to ensure that all generated grasps are valid under the model, i.e., belongs to the set of grasps shown by the teacher. Second, the regression aims not only to allow the robot to adapt to changing contact signature, but also to prevent the robot from behaving too far from what has been shown. For these reasons, it is necessary for the model to forbid the generation of posture and contact patterns that are too different from the examples in the training dataset.

In order to determine if query point $\boldsymbol{\phi}$ is likely under the model, we define a similarity function $f(\boldsymbol{\phi}, \boldsymbol{\alpha})$ that assigns to each point $\boldsymbol{\phi}$ of the input space, with reliability $\boldsymbol{\alpha}$, a membership value which is given by

$$f(\boldsymbol{\phi}, \boldsymbol{\alpha}) \quad = \quad \sum_{k=1}^{K} \overline{\mathcal{N}}\left(\boldsymbol{\phi}; \boldsymbol{\mu}_{\boldsymbol{\phi},k}, \tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\phi}\boldsymbol{\phi},k}(\boldsymbol{\alpha})\right) \quad (5)$$

where $\overline{\mathcal{N}}$ is derived from a normal distribution whose output has been normalized between 0 and 1, i.e.

$$\overline{\mathcal{N}}\,(x; \boldsymbol{\mu}, \boldsymbol{\Sigma}) \quad = \quad \exp\left(-\frac{1}{2}(x - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (x - \boldsymbol{\mu})\right)$$

In comparison to the marginal likelihood $p(\boldsymbol{\phi}, \boldsymbol{\alpha})$, this membership function has the advantage of considering each Gaussian component to have the same importance, irrespectively of the proportion of datapoints that have been used to train each components.[10] This effect is the result of (i) the absence of the priors in Eq. 5, and (ii) the normalization of $\overline{\mathcal{N}}$. With respect to the second reason, a point ϕ that is located within a given distance (in the Mahalanobis sense) of a Gaussian will receive the same value, irrespective of the size of its covariance.

We then search for $\boldsymbol{\phi}^\star$, the closest point to the current query point $\boldsymbol{\phi}$, that has a membership value $f(\boldsymbol{\phi}^\star, \boldsymbol{\alpha})$ higher than a given threshold η.[11]. In our previous work [42], we used a closed-form solution to this problem, since for a given point

[9]In practice, we did not consider the likelihood in its strict sense. As will be described in the next paragraphs, we rather consider a membership function that is derived from the likelihood. It has the main advantage of cancelling the effect of the variable proportion of data-points used to train each component of the GMM.

[10]Because of the nature of our data collection paradigm, i.e., human demonstration, several feasible grasp may be shown more often than others. Learning from such a non-uniform dataset induces a bias into the priors p_k of each component of the mixture, which may compromise the selection of grasps that were shown less frequently by the teacher.

[11]To fix the threshold η in our experiments, we consider that a point $\boldsymbol{\phi}$ belongs to the model if its Mahalanobis distance to any component of the GMM is below $\beta = 2$ standard deviations, which corresponds to $\eta = \exp(-1/2\,\beta^2)$.

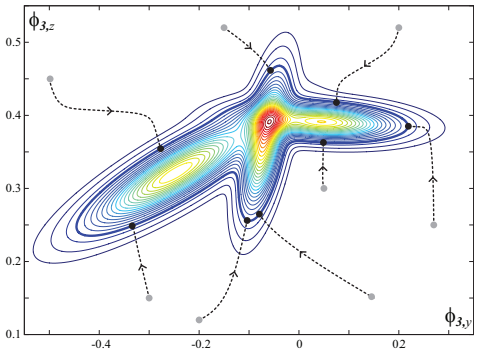

Fig. 5. Two-dimensional illustration of gradient ascent on the membership function for several input query points $\boldsymbol{\phi}$ with the can object. Light dots correspond to initial query inputs $\boldsymbol{\phi}$, dark dots to valid query inputs $\boldsymbol{\phi}^\star$, contours to parts of the space with constant membership value $f(\boldsymbol{\phi}, \boldsymbol{\alpha} = \mathbf{1})$, and the thick contour to threshold value η. Shown for two dimensions (y, z) of the contact signature for the index finger (ϕ_3).

in the input space, only a single Gaussian component was considered at a time. In our current formulation, a single point in the input space lies within a mixture of Gaussians, and so there exists no closed-form solution. We therefore adopt an iterative method. Given threshold η, if $f(\boldsymbol{\phi}, \boldsymbol{\alpha}) < \eta$ we perform gradient ascent on the membership function, until $\boldsymbol{\phi}^\star$ is found. The gradient of this function is given by

$$\frac{\partial}{\partial \boldsymbol{\phi}} f(\boldsymbol{\phi}, \boldsymbol{\alpha}) = \sum_{k=1}^{K} \overline{\mathcal{N}}\left(\boldsymbol{\phi}; \boldsymbol{\mu}_{\boldsymbol{\phi},k}, \tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\phi}\boldsymbol{\phi},k}(\boldsymbol{\alpha})\right) \left(\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\phi}\boldsymbol{\phi},k}(\boldsymbol{\alpha})\right)^{-1} (\boldsymbol{\phi} - \boldsymbol{\mu}_{\boldsymbol{\phi},k}) \quad (6)$$

and an illustration of the result of this procedure is shown in Figure 5. Note that for $f(\boldsymbol{\phi}, \boldsymbol{\alpha}) \geq \eta$, gradient ascent is unnecessary and hence $\boldsymbol{\phi}^\star = \boldsymbol{\phi}$.

2) Model Prediction via Regression: Once a valid contact signature input $\boldsymbol{\phi}^\star$ is obtained, the next step is to estimate the desired posture $\hat{\boldsymbol{\theta}}$ and pressure $\hat{\boldsymbol{s}}$ for the fingers. Thus, we compute the conditional of our joint probability model by means of Gaussian Mixture Regression [43], which gives

$$p(\boldsymbol{\theta}, \boldsymbol{s} \,|\, \Omega, \boldsymbol{\phi}^\star, \boldsymbol{\alpha}) \sim \mathcal{N}\left(\begin{pmatrix} \hat{\boldsymbol{\theta}} \\ \hat{\boldsymbol{s}} \end{pmatrix}, \begin{pmatrix} \hat{\boldsymbol{\Sigma}}_{\boldsymbol{\theta}\boldsymbol{\theta}} & \hat{\boldsymbol{\Sigma}}_{\boldsymbol{\theta}\boldsymbol{s}} \\ \hat{\boldsymbol{\Sigma}}_{\boldsymbol{s}\boldsymbol{\theta}} & \hat{\boldsymbol{\Sigma}}_{\boldsymbol{s}\boldsymbol{s}} \end{pmatrix}\right) \quad (7)$$

with expectation

$$\begin{pmatrix} \hat{\boldsymbol{\theta}} \\ \hat{\boldsymbol{s}} \end{pmatrix} = \sum_{k=1}^{K} \beta_k(\boldsymbol{\phi}^\star, \boldsymbol{\alpha}) \left[\begin{pmatrix} \boldsymbol{\mu}_{\boldsymbol{\theta},k} \\ \boldsymbol{\mu}_{\boldsymbol{s},k} \end{pmatrix} + \begin{pmatrix} \boldsymbol{\Sigma}_{\boldsymbol{\theta}\boldsymbol{\phi},k} \\ \boldsymbol{\Sigma}_{\boldsymbol{s}\boldsymbol{\phi},k} \end{pmatrix} \left(\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\phi}\boldsymbol{\phi},k}(\boldsymbol{\alpha})\right)^{-1} \left(\boldsymbol{\phi}^\star - \boldsymbol{\mu}_{\boldsymbol{\phi},k}\right)\right]$$

and expected variance

$$\begin{pmatrix} \hat{\boldsymbol{\Sigma}}_{\boldsymbol{\theta}\boldsymbol{\theta}} & \hat{\boldsymbol{\Sigma}}_{\boldsymbol{\theta}\boldsymbol{s}} \\ \hat{\boldsymbol{\Sigma}}_{\boldsymbol{s}\boldsymbol{\theta}} & \hat{\boldsymbol{\Sigma}}_{\boldsymbol{s}\boldsymbol{s}} \end{pmatrix} = \sum_{k=1}^{K} \beta_k^2(\boldsymbol{\phi}^\star, \boldsymbol{\alpha}) \left[\begin{pmatrix} \boldsymbol{\Sigma}_{\boldsymbol{\theta}\boldsymbol{\theta},k} & \boldsymbol{\Sigma}_{\boldsymbol{\theta}\boldsymbol{s},k} \\ \boldsymbol{\Sigma}_{\boldsymbol{s}\boldsymbol{\theta},k} & \boldsymbol{\Sigma}_{\boldsymbol{s}\boldsymbol{s},k} \end{pmatrix} - \begin{pmatrix} \boldsymbol{\Sigma}_{\boldsymbol{\theta}\boldsymbol{\phi},k} \\ \boldsymbol{\Sigma}_{\boldsymbol{s}\boldsymbol{\phi},k} \end{pmatrix} \left(\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\phi}\boldsymbol{\phi},k}(\boldsymbol{\alpha})\right)^{-1} \begin{pmatrix} \boldsymbol{\Sigma}_{\boldsymbol{\theta}\boldsymbol{\phi},k} \\ \boldsymbol{\Sigma}_{\boldsymbol{s}\boldsymbol{\phi},k} \end{pmatrix}^T\right]$$

$$\beta_k(\boldsymbol{\phi}^\star, \boldsymbol{\alpha}) = \frac{p(k)\,\mathcal{N}\left(\boldsymbol{\phi}^\star; \boldsymbol{\mu}_{\phi,k}, \tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\phi}\boldsymbol{\phi},k}(\boldsymbol{\alpha})\right)}{\sum_{i=1}^{K} p(i)\,\mathcal{N}\left(\boldsymbol{\phi}^\star; \boldsymbol{\mu}_{\boldsymbol{\phi},i}, \tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\phi}\boldsymbol{\phi},k}(\boldsymbol{\alpha})\right)}$$

where $\beta_k(\boldsymbol{\phi}^\star, \boldsymbol{\alpha})$ is the posterior probability of the k^{th} component responsible for the query input $\boldsymbol{\phi}^\star$ with reliability $\boldsymbol{\alpha}$.

Here, we can observe the effect of the reliability measure $\boldsymbol{\alpha}$ on the regression. For unreliable contact pressure readings, i.e. $\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\phi}\boldsymbol{\phi},k}(\boldsymbol{\alpha}) \to \infty$, the conditional will simply ignore the contribution of $\boldsymbol{\phi}^\star$, and thus output the mean hand posture and contact sensor reading of the model. In contrast, for reliable pressure readings, i.e. $\tilde{\boldsymbol{\Sigma}}_{\boldsymbol{\phi}\boldsymbol{\phi},k}(\boldsymbol{\alpha}) \to \boldsymbol{\Sigma}_{\boldsymbol{\phi}\boldsymbol{\phi},k}$, the conditional becomes equivalent to GMR on the original GMM. The same principle applies if one or more fingers are no longer in contact.

3) Finger Actuation: To control finger actuation and achieve the targets produced by the model, we define a feedback controller that takes as input the error between target and current grasp configuration. Since in practice it is often not possible to satisfy both position and force constraints simultaneously, we design our controller to blend the minimization of both error signals in a continuous manner.

The general idea behind our controller is to give priority to position control, so that force control is progressively activated as the current posture gets in the vicinity of the target posture. To get an estimate of how near the robot is to the desired posture, we compute a positional error measure $\lambda \in [0..1]$ that is weighted by the inverse of the covariance of the conditional along the dimensions of the hand pose,

$$\lambda = \exp\left(-\frac{1}{2}(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})^T\, \hat{\boldsymbol{\Sigma}}_{\boldsymbol{\theta}\boldsymbol{\theta}}^{-1}\, (\hat{\boldsymbol{\theta}} - \boldsymbol{\theta})\right) \quad (8)$$

In detail, position control is handled by a Proportional-Integral-Derivative (PID) controller minimizing the error in hand pose $\boldsymbol{e}_\theta = \hat{\boldsymbol{\theta}} - \boldsymbol{\theta}$, and force control is handled by a Proportional-Derivative (PD) controller using the target contact sensor value $\hat{\boldsymbol{s}}$ and its error $\boldsymbol{e}_s = \hat{\boldsymbol{s}} - \boldsymbol{s}$. Blending between the two controllers is accomplished via factor λ, such that

$$\boldsymbol{u}\left(\hat{\boldsymbol{\theta}}, \boldsymbol{\theta}, \hat{\boldsymbol{s}}, \boldsymbol{s}\right) = (1-\lambda)\left(\kappa_\theta^P\, \boldsymbol{e}_\theta + \kappa_\theta^D\, \dot{\boldsymbol{e}}_\theta + \kappa_\theta^I \int \boldsymbol{e}_\theta \,\mathrm{d}t\right) + \lambda \boldsymbol{M}\left(\kappa_s^C\, \hat{\boldsymbol{s}} + \kappa_s^P\, \boldsymbol{e}_s + \kappa_s^D\, \dot{\boldsymbol{e}}_s\right) \quad (9)$$

where $\dot{\boldsymbol{e}}_\theta$ and $\dot{\boldsymbol{e}}_s$ are the time derivative of the error in position and contact sensor reading, respectively. The first term of the equation handles the position PID control, and the second term the force PD control. Thus when far from the target posture $(\lambda \to 0)$ position control is employed, and when near the target posture $(\lambda \to 1)$ force control takes over.

The matrix $M \in \mathbb{R}^{N_\theta} \times \mathbb{R}^{N_s}$ maps the signal of the force controller to the motors of each joint responsible for minimizing the contact sensor reading error of each finger. Each element M_{ij} of this matrix is given by:

$$M_{ij} = \begin{cases} 1 & \text{if the } j^{\text{th}} \text{ contact sensor is mounted on the same finger than the } i^{\text{th}} \text{ motor} \\ 0 & \text{otherwise} \end{cases}$$

Finally, $\kappa_\theta^P, \kappa_\theta^D, \kappa_\theta^I, \kappa_s^C, \kappa_s^P, \kappa_s^D$ are the gains of our controller.[12]

[12]In our implementation these gains are tuned by hand ($\kappa_\theta^p = 60,\ \kappa_\theta^d = 7,\ \kappa_\theta^i = 2,\ \kappa_s^c = 30,\ \kappa_s^p = 25,\ \kappa_s^d = 5$).Note that these gains operate on values of current, which are then mapped to torque commands for the motors.

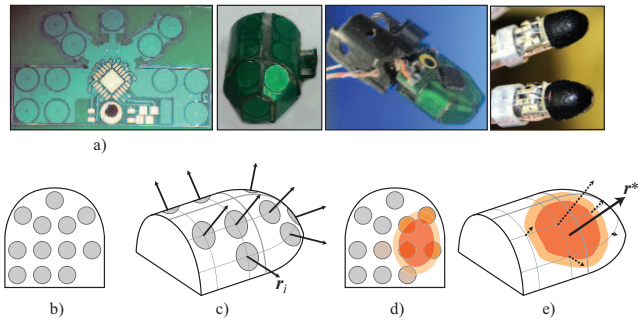

Fig. 6. Fingertip sensor technology (a). Each sensor of a given fingertip (b) is associated with a unit vector $\boldsymbol{r}_i$, normal to the finger surface at the sensor location (c), whose magnitude is scaled by the sensor response (colored surface) (d) when estimating contact normal $\boldsymbol{r}^*$ (e).

IV. Implementation Details

The implementation details of our empirical validation are provided here, with regards to robot and demonstration hardware, as well as to task domain and evaluation.

A. Robot Platform and Tactile Sensors

Our approach is validated on a human-child sized 53-DoF robot, the *iCub* [14]. The hand of the iCub (Fig. 2) has 8 controllable DoF. Each finger, as well as the thumb, consists of 3 phalanges. A single motor controls the 2 joints between the 3 phalanges with a single cable, and an additional motor controls the joint between the digit and hand, for a total of 2 controllable DoF per digit. Control of the ring and pinky fingers is coupled. Finally, an additional motor controls the opposition of the thumb. The tendon system of the robot hand allows for a small amount of compliance in the first joint, and a larger amount in the second joint (towards the fingertip) of each digit.

A fingertip sensor array (14.5mm $\times$14mm) is mounted at the end of each finger and thumb (Fig. 6a). The array consists of 12 capacitive pressure sensing nodes and the electronics processing for the A/D conversion [47]. The fingertip is made of compliant and deformable silicone patches whose capacitance varies when pressure is applied at the surface. Consequently, with this array it is possible to measure contact direction and magnitude at different locations on the fingertip.

In the experimental work presented in this paper, the thumb, index finger and middle finger are utilized, but not the two coupled digits.[13] The pose of the hand therefore consists of the joint angles for each of the 2 controllable degrees of freedom in the 3 digits utilized, plus the joint that controls thumb opposition, and so $\boldsymbol{\theta} \in \mathbb{R}^7$. We define the sensor reading $\boldsymbol{s} \in \mathbb{R}^3$ as a vector containing a single real value for each fingertip $j \in \{1..3\}$, taken as the summed response over all sensor nodes $p_{j,i}, i \in \{1..12\}$ on the given fingertip, i.e. $s_j = \sum_i p_{j,i}$. We further define the contact signature $\boldsymbol{\phi} \in \mathbb{R}^9$ as a vector containing an estimate of the contact normal on each fingertip. As illustrated in Figure 6 (c,e), for each sensor i on a fingertip, we define a direction $\boldsymbol{r}_i \in \mathbb{R}^3$ corresponding to the unit vector normal to the finger surface at the sensor location. From this, we compute the global directional response[14] $\boldsymbol{r}_j^\star$ of each fingertip j as the sum of these normals weighted by the response of each sensor, i.e. $\boldsymbol{r}_j^\star = \sum_i p_{j,i}\, \boldsymbol{r}_i$. To obtain the three-dimensional contact direction $\boldsymbol{\phi}_j$, we normalize the global response, i.e., $\boldsymbol{\phi}_j = \boldsymbol{r}_j^\star / \|\boldsymbol{r}_j^\star\|$.

[13]The choice of not using the two last digits is motivated by their tight coupling. A single motor controls the motion of both fingers, and this underactuation makes them difficult to use for fine manipulation tasks.

B. Demonstration Interface

Demonstration is performed via teleoperation by a human teacher, who simultaneously controls the 7 DoF of the iCub hand. Teleoperation is accomplished through a joint recording system and a mapping that allows the human to directly control the motion of the robot hand by moving her own hand, during which the robot records from its own sensors.[15] The data glove (Fig. 1, top) worn by the teacher contains 14 torsion sensors that detect the angle of the joints in the human hand. We then map the human joint angles to the joint angles of the robot hand, thus accomplishing remote control. A key limitation of this teleoperation interface is the absence of haptic feedback for the human, making the demonstration of a satisfactory level of contact - this is neither too strong nor too weak - difficult to estimate.

C. Validation Task

Task models are built for multiple objects, beginning with a single demonstration and following with 2 rounds of refinement via repeated correction-replay-learning steps. We refer to these models as the *Demonstration-Refinement-Refinement (DRR)* models. Specifically, task models are built for the following 4 objects (Fig. 7): a small cylindrical can (5.7cm diameter, 14.6cm height), a large cylindrical can (6.5cm diameter, 11.7cm height), a box (6.0cm $\times$6.0cm $\times$3.0cm) and a straightedge ruler (1.4cm $\times$31.6cm). Each model is learned 3 times.

Reused models also are built for multiple objects, beginning with an existing model and following this with 1 round of refinement when interacting with the new object. We refer to these models as the *reUse-Refinement (UR)* models. Reused models are built for and from the following objects: for the small can from the big can (different size, similar shape), for the box from the small can (similar size, different shape) and for the big can from the box (different size, different shape). Each task model developed from model reuse again is learned 3 times.

During correction, the can objects are perturbed by pulling a can side to side, in sweeps that run parallel to the length of the robot palm, as well as pitching the can forwards and backwards (Fig. 7, left). The box is perturbed in a similar fashion. The ruler is held vertically and perturbed by pitching it to the either side (Fig. 7, right).

[14]For our experiments, considering the existence of a single unimodal pressure zone for each fingertip is a fair assumption.

[15]Note that the mechanism used to provide tactile corrections - by gently pulling or pressing on the fingers - is only able to move the fingers within their compliance limits for a given posture. Transitioning to a sufficiently different posture, like the transition from an open to partly closed hand, must be achieved through another mechanism, for example teleoperation.

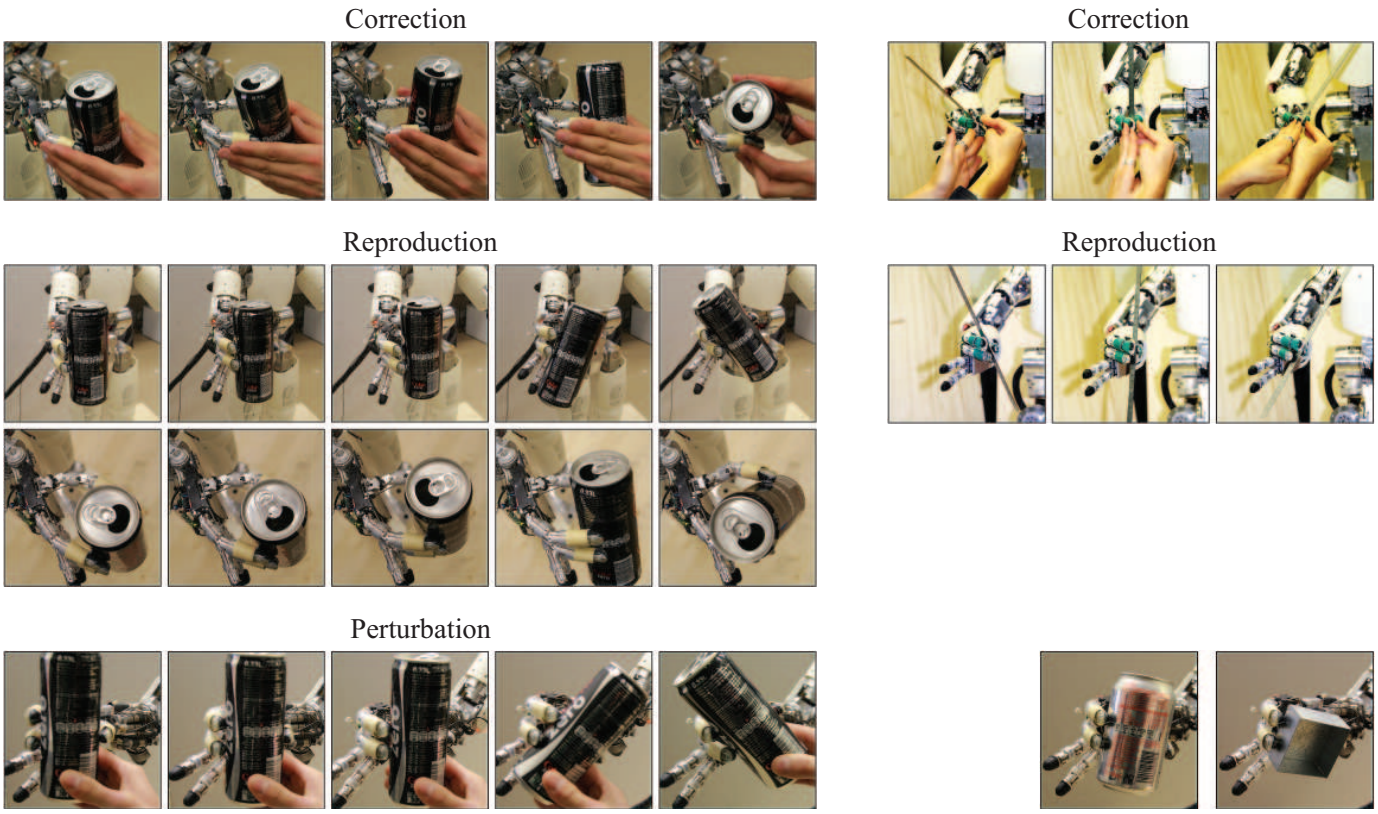

Fig. 7. Tactile correction for learning grasp adaptation. The teacher indicates adaptability within the compliance constraints of a hand posture (*Correction*). The learner then replays the sequence of corrected poses (*Reproduction*). The final learned model is able to adapt the pose in response to different contact signatures (*Perturbation*). Objects: small can (left panels), ruler (right, top), large can and box (right, bottom).

D. Task Evaluation

We assess the performance of our approach by evaluating the evolution of the quality of the models across each learning phase. During the development of DRR models, we compare the execution quality along the following model progression:

$$\Omega_0 \rightarrow \tilde{\Omega}_1 \rightarrow \Omega_1 \rightarrow \tilde{\Omega}_2 \rightarrow \Omega_2$$

where Ω_0 is the model derived from the initial demonstration, $\tilde{\Omega}_1$ follows immediately correction of the initial model Ω_0 *before* self-demonstration. $\tilde{\Omega}_1$ was thus learned using *unreliable* sensory data gathered directly during the correction phase. Ω_1 is the result of one full correction cycle, $\tilde{\Omega}_2$ follows correction of the model Ω_1 *before* self-demonstration, and Ω_2 is the final model obtained after the second full correction cycle.

In order to demonstrate the efficiency of the model reuse paradigm, we compare the quality of three models. First, we consider the immediate reuse of the model Ω_2^A, learned on object A after 2 rounds of correction when applied on a novel object B. Then we *refine* this reused model by performing a complete correction cycle using object B, producing a model that we denote as $\Omega_1^{B_A}$. We then test it on object B. Finally, we compare the latter's performance with the model Ω_1^B that was previously learned from scratch via demonstration with object B.

To evaluate each model, the controller and model are run while having the teacher physically perturb the object to explore the full range of possible hand poses that the model can handle. The teacher also pushes the object within the robot fingers' compliance limits, past the postures predicted by the model. During this evaluation, we gather a sequence of samples $\{(\boldsymbol{\phi}^i, \boldsymbol{\theta}^i, \boldsymbol{s}^i, \hat{\boldsymbol{\theta}}^i, \hat{\boldsymbol{s}}^i)\}_{i=1}^N$ at a rate of 20ms for a total duration of about 15 seconds. From all of the samples, we only retain those that are considered to belong to the model, according to the same criteria used to estimate nearness to a target posture when blending position and force control (Eq. 8). The following metrics are then used to evaluate model performance for each testing set:

- *Range of Motion*: The difference between the minimum and maximum joint angle values for each finger:
  $\max_{i=1..N} \theta^i - \min_{i=1..N} \theta^i$.
  In order to reduce the number of variables to analyze, we combine the range values into 4 groups. We consider the sum of both joint angles for each of the 3 fingers (3 groups), and separately, the thumb opposition angle (1 group).
- *Time in Force-Closure*: The percentage of time where the three fingers are in contact with the object and the resulting grasp attains force closure [15]:
  $\frac{1}{N}\sum_{i=1}^{N}((\boldsymbol{s}^i > 0) \text{ and } (FC(\boldsymbol{\phi}^i, \boldsymbol{\theta}^i) > 0))$
  The force-closure function $FC(\boldsymbol{\phi}, \boldsymbol{\theta}) \in \{0, 1\}$ is computed using the method described in [48].
- *Contact Error*: The difference between the target (model-

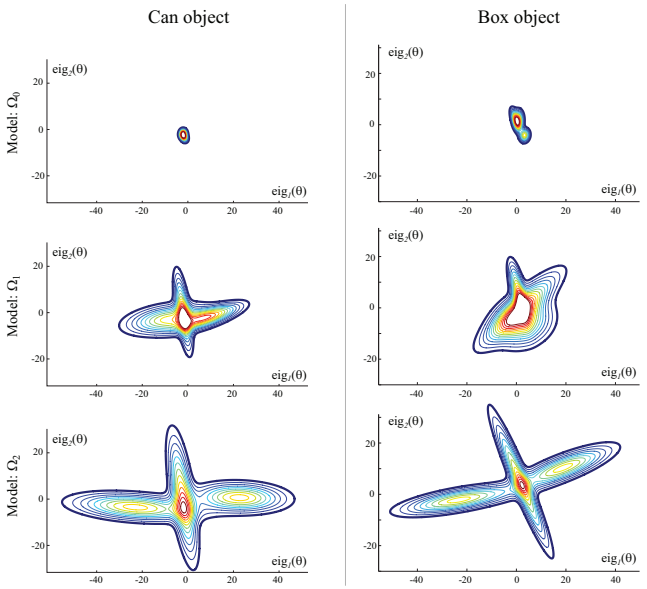

Fig. 8. After several rounds of refinement, there is an increase in the range of motion that a model has learned and hence, can use for grasp adaptation. Example data is given for two different objects (columns). The axes correspond to the projection of the joint space $\theta \in \mathbb{R}^7$ on the first two principal components. Contours correspond to parts of the space with constant marginal likelihood value $p(\boldsymbol{\theta}|\Omega)$, given a learned task model Ω.

predicted) and actual (controller-executed) contact values, averaged across all timesteps of the perturbation:
$\frac{1}{N}\sum_{i=1}^{N} \|\boldsymbol{s}^i - \hat{\boldsymbol{s}}^i\|_1$
where $\|\boldsymbol{x}\|_1$ denotes the L1-norm of $\boldsymbol{x}$.

- *Shakiness*: The difference between the raw and smoothed joint velocities, averaged across the testing period:
$\frac{1}{N}\sum_{i=1}^{N}(|\dot{\theta}^i| - |\dot{\tilde{\theta}}^i|)$
where the smoothed velocity $\dot{\tilde{\theta}}^i$ is computed via windowed averaging over $\dot{\theta}^i \in \chi$ (window size = $0.4s$).

The *Time in Force-Closure* measure provides an indication of grasp stability and adaptation quality, where higher means with lower variances suggest constant contact with the object and thus, efficient grasp adaptation. The *Contact Error* measure relates to how well the model provides appropriate adaptation inputs for the controller, where low error corresponds to the controller regularly being able to attain the predicted values (i.e. smooth adaptation commands). The *Range of Motion* measure points to the responsiveness of the learned model, with a high value indicating adaptation over a large range of hand postures. The *Shakiness* measure highlights instances of jerky or sudden movements, via high values that indicate a sharp change in joint angle velocity.

V. Results

This section presents the findings of our empirical evaluations. Task models for multiple objects were successfully built, refined and reused with our approach.

A. Model Refinement

To look specifically at the effect of refinement, comparisons will be made between models Ω_0, Ω_1 and Ω_2, on each of the four objects. Furthermore, the necessity of self-replay is highlighted, with the comparison of models $\tilde{\Omega}_1$ vs. Ω_1, and $\tilde{\Omega}_2$ vs. Ω_2.

1) Larger Joint Angle Ranges: Tactile corrections were able to provide to the models a broader range of postures that enabled grasp adaptation. Figure 9 reports the *range of motion* averaged across all objects for each phase of our teaching process, while detailed results for each object are given in Table I. A significant trend of increased range of motion was seen across objects, for all joints (p-value $p < 0.001$, two-way ANOVA[16]) as well as within objects ($p < 0.05$, two-way Student t-test [17]) Moreover, this trend continued with an additional round of refinement, as the models of all objects, except those of the ruler, displayed their largest ranges after the second round of correction-replay-learning. This difference in behavior across objects was marked by the significance of the interaction term of the ANOVA, i.e., the object type was an important factor in explaining the statistical results. Nevertheless, removing the ruler from the testing dataset canceled this interaction. The reason is that the range of finger postures with which the ruler object can be grasped is small, and thus can be demonstrated in a single refinement cycle.

Figure 8 shows two examples that illustrate the growth of the region of the joint-space that has been learned after each round of correction.

2) More Stable Contact: Figure 9 also reports the time in force-closure averaged across all objects. This time significantly increased with one round of refinement (Ω_0 vs. Ω_1) for half of the objects, as well as across objects ($p < 0.001$). This measure however appeared to stabilize after one round of refinement, and did not really improve with a second round (Ω_1 vs. Ω_2). However, given that the range of motion displayed by the models drastically increased throughout each round of correction, the important result is that the time in force-closure did not decrease. Paired with the observation that the variance slightly reduced, these data suggest that, as a result of refinement, the grasps produced by the adaptation mechanism are more stable. Such a conclusion is further supported by the contact error data (Fig. 9), which significantly reduced with refinement across all objects ($p < 0.001$). The model thus more consistently made predictions that were appropriate for the controller.

However, the transitions between hand poses were not found to become significantly smoother with refinement, as indicated by the shakiness measure in the average over objects (Fig. 9). Nevertheless, this trend we expect is also related to the explored range of hand poses, which increased at each

[16]A two-way ANOVA using factors F_1: *object type* ={*small can, big can, box, ruler*} and F_2: *training phase*={Ω_i, Ω_j} was performed on selected pairs i and j of training phases. Our testing data sample consists of 3 repeated measures for each category.

[17]The Student t-test was performed by comparing the results obtained between the selected training phases for each object separately. The small number of samples (3 repeated measures per object and training phase) motivates our choice of considering a higher p-value for significance.

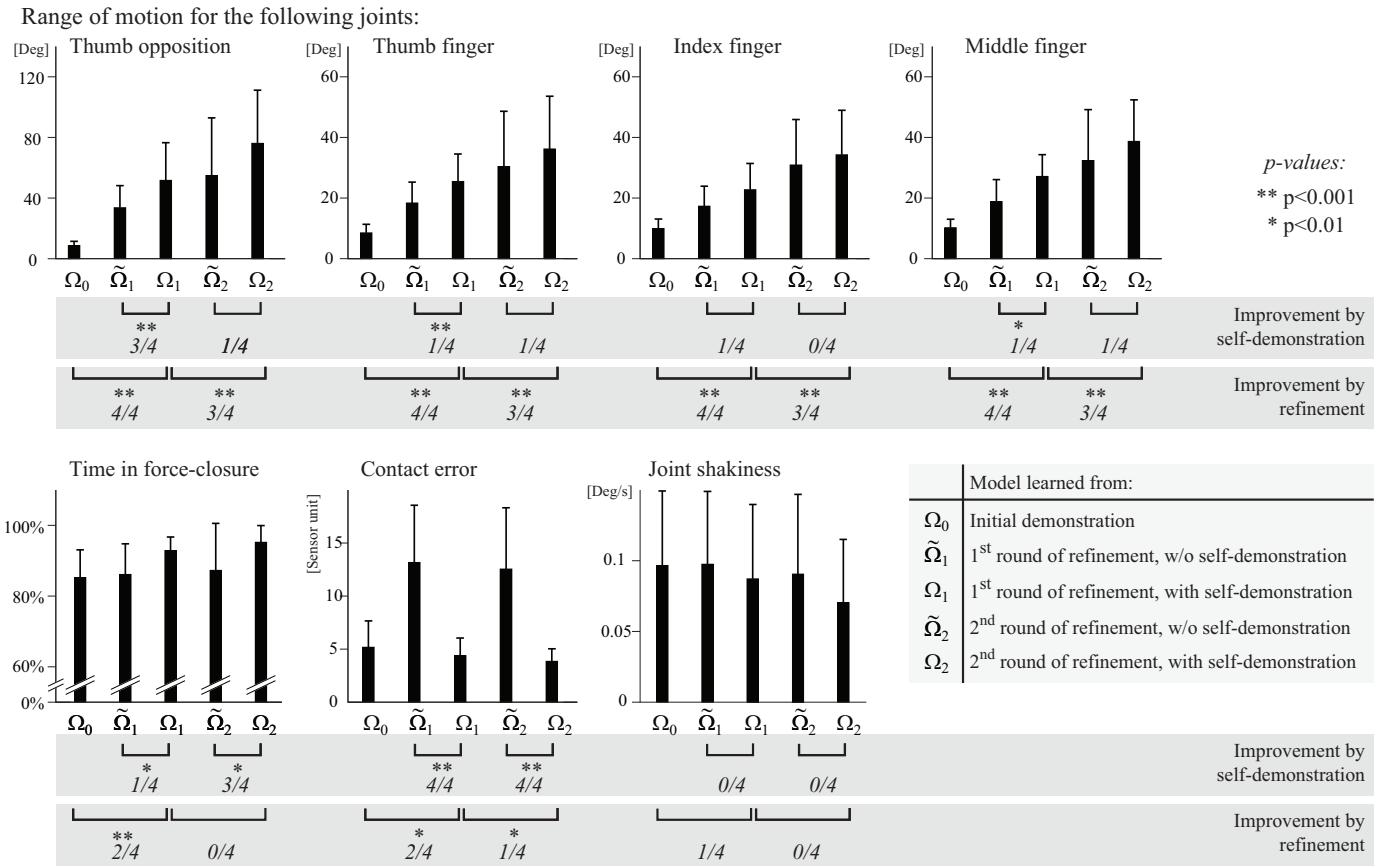

Fig. 9. Evaluation of policy *refinement*: Range of motion, time in force-closure, contact error and shakiness measures are given for each model produced during the development of the *Demonstration-Refinement-Refinement* (DRR) models. (Average values across 3 repetitions of the experiments for all objects.) Below each plot, stars indicate significant improvements between training phases. Ratios reports the number of objects that, taken separately, exhibited an significant improvement across each phase ($p < 0.05$).

correction cycle. Again, the important result is here that the shakiness measure did not increase.

3) The Utility of Self-Demonstration: For all object models, an increase in performance according to almost all measures was observed following self-demonstration compared to the model derived following tactile correction (Fig. 9, $\tilde{\Omega}_1$ vs. Ω_1, $\tilde{\Omega}_2$ vs. Ω_2). Although these performance increases were not generally significant for the joint ranges, the time in force-closure and the contact error measures showed a significant increase. This confirms our hypothesis that the additional contact of the teacher's hands does in fact add noise to the data, and that a more accurate contact signature is gained through learner replay of the corrected hand postures.

B. Model Reuse

To look at the effect of reuse, comparisons will be made between the models Ω_2^A learned for an object A and reused on a novel object B, the subsequently refined models $\Omega_1^{B_A}$, and the DDR models Ω_1^B learned for object B, for multiple combinations of objects A and B.

The main motivation for model reuse comes from the fact that two models, learned for two similar objects, may have a lot in common. Therefore, rather than re-learning the grasp adaptation task from the beginning for each novel object, model reuse takes advantage of the information that has been acquired previously. To illustrate this argument, Figure 11 shows the model encoding for the contact signature data of two different objects. As can be seen, the areas covered by each model include a lot of overlap, and thus the reuse of the knowledge encapsulated in the first model will likely bootstrap the learning of the second one.

1) Effective Transfer of Joint Angle Domain Knowledge: The range of motion averaged across each UR model is provided in Figure 10, and detailed values are given in Table II. Here we note that the range values achieved following reuse are similar to those seen after demonstration plus one round of refinement (Ω_2^A vs. Ω_1^B). When reusing the model Ω_2^A, given that no effort has yet been invested into model learning, and that by contrast, the DRR model Ω_1^B has already undergone demonstration plus one round of refinement, these data highlight the utility of model reuse as an effective means for transferring domain knowledge and reducing the effort involved in model development.

After refining models Ω_2^A, the range of motion further expanded slightly (Ω_2^A vs. $\Omega_1^{B_A}$). In addition, the positive difference in range of motion observed between the UR models $\Omega_1^{B_A}$ and the DRR models Ω_1^B was highly significant for the thumb opposition joint ($p < 0.001$), but less for the other joints (within and across tested objects). This result can primarily be explained by the importance of this joint for producing a

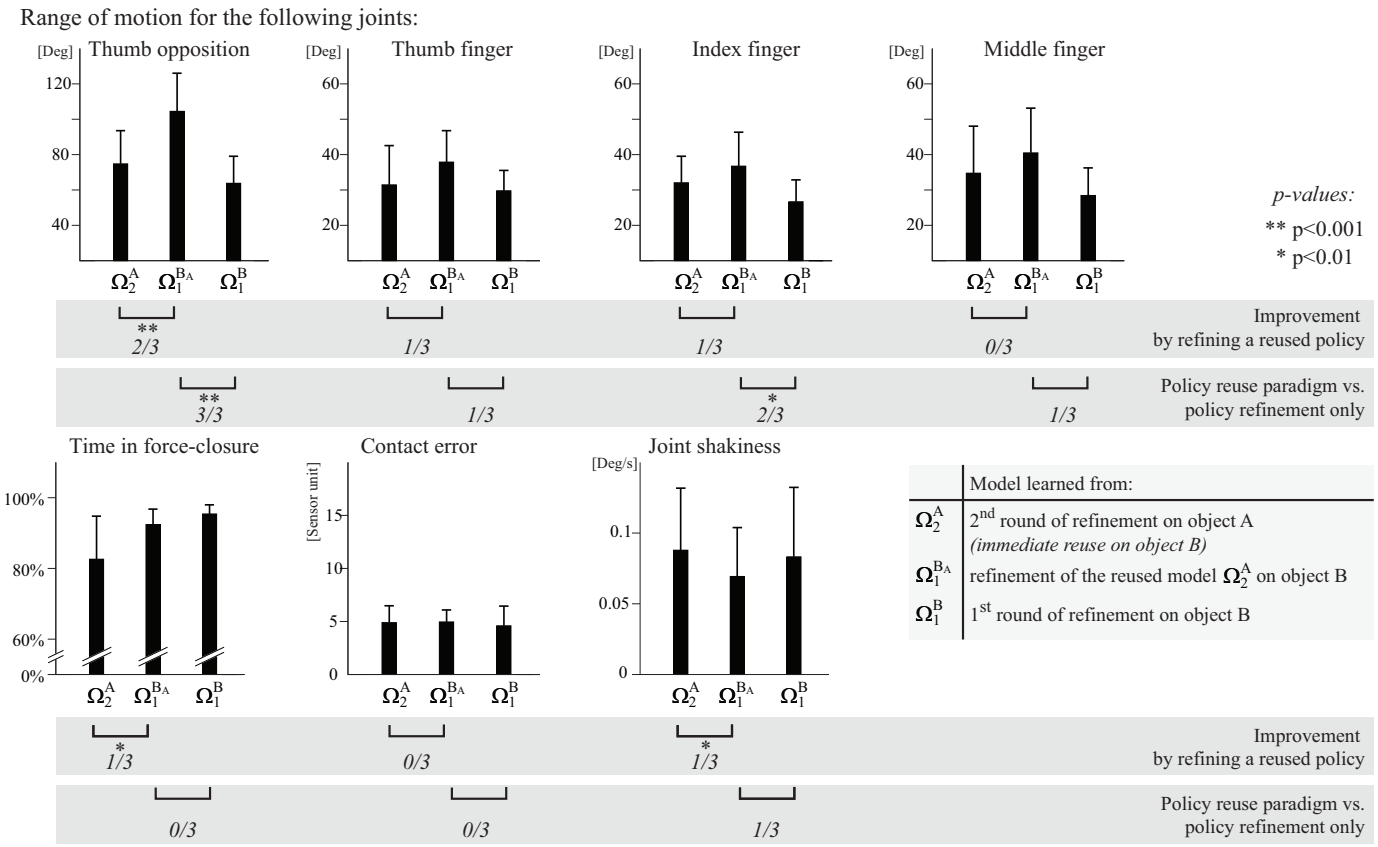

Fig. 10. Evaluation of *policy reuse*: Range of motion, time in force-closure, contact error and shakiness measures are given for each model considered for evaluating the policy reuse paradigm. (Average values across 3 repetitions of the experiments for all objects.) Below each plot, stars indicate significant improvements between training phases. Ratios reports the number of objects that, taken separately, exhibited an significant improvement across each phases ($p < 0.05$).

larger variety of valid grasps within our experimental setup. All together, these data support our hypothesis that model reuse is an effective means of transferring domain knowledge.

2) Contact and Smoothness of Adaptation: The desirable high values for the time in force-closure (Fig. 10) did diminish following immediate reuse. This can easily be explained by the dissimilarity of the contact signature between the different objects, producing less appropriate prediction signals. Nevertheless, performance then significantly improved across all objects following a round of refinement (Ω_2^A vs. $\Omega_1^{B_A}$, $p < 0.001$), with final values approaching those of the DRR models across all objects ($\Omega_1^{B_A}$ vs. Ω_1^B).

The trend of effective domain knowledge transfer with reuse was further underlined by the shakiness measure (Fig. 10), which displayed similar values for the initial UR models and DRR models following demonstration plus refinement (Ω_2^A vs. Ω_1^B). Importantly, this measure improved with refinement on average for all models (Ω_2^A vs. $\Omega_1^{B_A}$). The immediate reuse of a model for another object having a different contact signature produced less reliable contact information, and thus worse control of the fingers. However, refining the model on the correct object overcame this effect.

C. Grasp Execution: Reliability Measure

To look at the effect of augmenting the model by incorporating a prior on the reliability of the sensor signal (see Section III-C2), we compared the performance of augmented and non-augmented models, by using the models learned for each object after two rounds of refinement. As the experimental conditions of the previously described experiments did not produce a large proportion of missing contacts, the effect of augmenting the models did not result in a significant improvement. In order to justify this part of our approach, we present here the results of another experiment, where we artificially corrupted the signal coming from a selected fingertip. To mimic the fact that, in the absence of contact, the touch sensors produce a default noisy response, we set the response

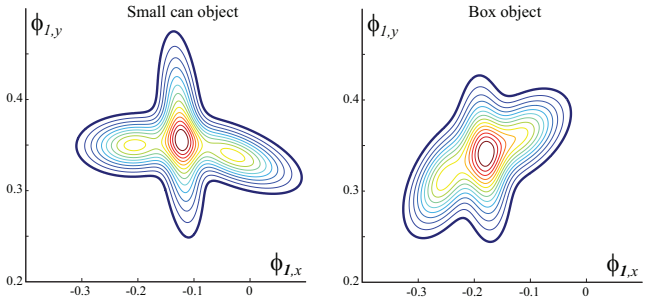

Fig. 11. Two-dimensional contact signature for the small can (left) and box (right) objects. Shown for two dimensions (x, y) of the contact signature for the thumb (ϕ_1). Contours correspond to parts of the space with constant membership function value $f(\boldsymbol{\phi}, \boldsymbol{\alpha} = \mathbf{1})$.

$p_{j,i}$ of the corrupted finger j to follow a normal distribution $\mathcal{N}(\mu_{\text{noise}}, \sigma_{\text{noise}})$, where μ_{noise} and σ_{noise} correspond to the mean and standard deviation measured on the response of the real sensors of the iCub.

For each type of model (augmented and non-augmented), we performed 3 repetitions of an adaptation trial where the same finger was corrupted. These trials were run for each possible finger, for a total of 9 trials per model type. We then performed a three-way ANOVA on our experimental data in order to separate the effects of object type, model type and which finger was corrupted. Our main results are summarized in Figure 12.

We observed a significant increase of performance for the time in force closure as the model was augmented with signal reliability measures ($p < 0.001$). This can be explained by the random contact signature that is generated by the corrupted fingertip. If it is not canceled out by assigning it a low reliability, the information it conveys participates equally to the selection of the grasp to which to adapt. Because of that noise, the target grasp may vary importantly, and therefore result in a less stable adaptation. Then, whereas the effect of the object type was not significant, the effect of the corrupted finger was also important. The reason for that comes from the arrangement of the considered grasps: the thumb on one side and the two other fingers on the other side. As such, loosing the signal on the thumb results in a greater loss of sensory information compared to the case where only the information provided by index or the middle fingertip was corrupted. These results were corroborated by the shakiness measure. Although this measure was in general higher than in conditions where sensors were not artificially corrupted (see Fig. 9), augmented models compensated better for a loss in contact ($p < 0.01$). We also observed a higher variability in the response of non-augmented models. Consequently, undesired finger movements were more likely to appear, hence yielding a higher shakiness. Finally, no significant change in the range of motion was observed as an effect of augmenting the model. This was expected since the range of joint angles value spanned in each condition was the same.

In summary, despite the fact that the sensory feedback of a finger was corrupted, the augmented models still managed to make robust predictions that kept the grasp in force-closure throughout the adaptation task.

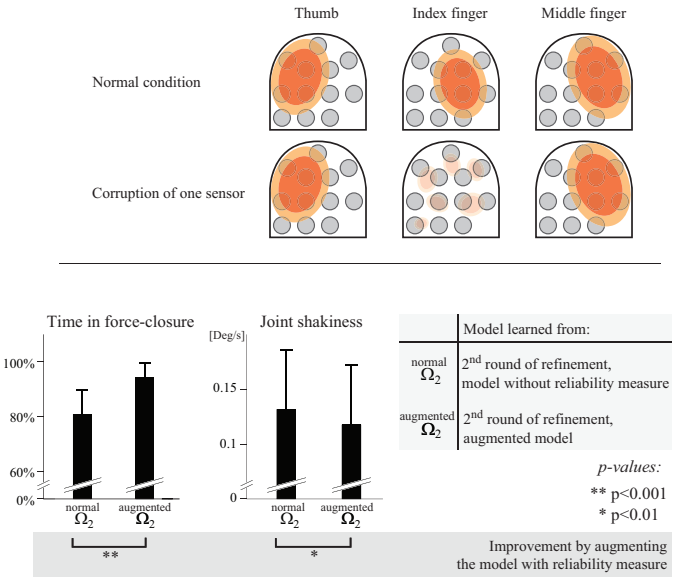

Fig. 12. Evaluation of the effect of augmenting the models with a *signal reliability measure*: *(top)* Illustration of the effect of corrupting the response of a selected fingertip: in this example, the middle finger. *(bottom)* Time in force-closure and shakiness measures are given for each type of model (Average values across 3 repetitions of the experiments for all objects and the corruption of each finger separately). Below each plot, stars indicate a significant difference between the performance of augmented versus non-augmented models.

VI. Discussion and Conclusion

We have introduced a probabilistic approach for grasp adaptation, which learns a model to adapt hand posture solely based on the sensor signature of the contact. A statistical model able to predict a target hand posture and contact magnitude, given the current contact normal direction, is learned from a dataset built over multiple steps under human supervision. In particular, an initial hand posture is first *demonstrated* to the learner, then physically *corrected* by a human teacher, and finally the resulting sequence of postures is *replayed* by the learner as a form of self-demonstration.

We contribute an empirical validation of our approach on the *iCub* robot. To provide tactile corrections, the teacher presses on the fingertips, thus exploiting partial compliance in the robot hand. Through this programming by demonstration methodology, we were able to teach a robot to perform the task by providing it not only with an implicit knowledge of the necessary kinematics for adaptation, but also with an intuitive notion of force. Our results confirmed successful grasp adaptation in response to changes in contact for multiple objects.

Our approach furthermore allows for the modification of a learned model, within two contexts. The first is to *refine* the model to improve adaptation performance, by repeating the correction-replay steps. The second is to *reuse* a model in the development of new model, characterized by the contact signatures of a different object. In both cases the teacher provides tactile corrections as the learner executes with an existing model of the task, thus exploiting the fact that corrections are easier to provide when the learner is already doing part of the job of actuation on its own, and building upon domain knowledge already present within the robot system. Both successful model reuse and improved adaptation with additional rounds of model refinement have been shown. Importantly, this iterative approach allowed us to progressively reduce the complexity of teaching the robot to perform a task that uses a large number of degrees of freedom.

The probabilistic task model that we learn is formulated to take advantage of the statistical data encoding in several important contexts. The first is to avoid over-generalization within the input space, by handling unreliable contact signature signals that might result from a missing contact between the object and one or more fingers, for example. The second is to follow a perturbation only when the hand is in a

posture that is near to what was seen within the demonstration dataset, and to otherwise counteract the perturbation in favor of maintaining posture stability. In short, the demonstration data thus is used not only to determine the reaction of the robot to environmental changes, but also to determine when grasps are infeasible or input signals are poor, by exploiting a probabilistic representation which captures the inherent variability in the data. A third advantage is to avoid the need of a detailed model of the hand kinematics and object geometry, by implicitly encapsulating this information into a model built from sensory data only. In contrast to model-based methods that require precise force sensing, actuation and a detailed environment model, which can be an impediment and impractical on many robotic platforms, our learning approach was capable of extracting the non-linearities inherent to such problems with a compact probabilistic model.

Our approach thus contributes to the challenging area of object interaction and manipulation within the context of dynamic environments, when contact with the object is changing due to large perturbations. Some limitations of this work include the following. The input space of our regression formulation is not sufficiently rich to disambiguate different hand postures that produce the same contact signature (i.e. contact normal direction ϕ), and so a model must be learned for each object individually. Also, the sensing capabilities of our robot platform have restricted our approach to the development fingertip manipulation paradigms only. A tactile sensor with greater coverage or finer resolution would allow for manipulations that engage the entire hand. Improving this sensory capability would also allow our approach to be applicable on a larger set of objects. A tactile sensor with greater coverage and resolution also might provide additional object information useful for defining an input space that is sufficiently rich to disambiguate different hand postures that produce the same contact signature. To tackle this latter issue, enhancing our prediction method to select the best grasp from a multi-modal distribution is a very interesting research question, that is left for future work.

Since our approach implicitly encapsulates the hand kinematics and object information, it is unlikely that a learned model would generalize directly to the addition or removal of one or more fingers. Nevertheless, models developed under our approach have been shown to be capable of handling the loss of sensory feedback from a finger. We therefore expect that one round of correction should be sufficient to learn, from the reuse of an existing model, a new model for a smaller number of fingers. If instead one or more fingers is added to the effector, the prior knowledge of the existing model would allow the teacher to focus on correcting the additional fingers only.

There are many other promising directions in which to continue this work. The first is to integrate the adaptive contact models with our prior work, that incorporated tactile corrections on the iCub arms, with the result of a complete tactile teaching interface for learning full hand-arm manipulation behaviors interactively via demonstration. One also might reason about the dynamics of the contact signatures, as they change over time. Integrating such information with the hand as well as arm posture adaptation would allow for increasingly complex responses to dynamic interactions with objects. For instance, our approach also assumes that the position of each finger on the object should remain roughly fixed throughout adaptation. Extending our work to incorporate finger repositioning techniques used for explicit object manipulation would certainly enhance the general applicability of our method.

At a more technical level, a more advanced model of finger actuation could be incorporated, for example that takes cable friction into consideration. We expect that an improved actuation model would have a significant impact on the success of the learned behavior, as the performance of a grasping system depends heavily on the actuation controller. Similarly, though the use of an impedance controller would require knowledge of the dynamic parameters of the manipulator and very precise force sensing capabilities, with such a controller our approach could be applied on a larger variety of robots, especially on those that do not have the intrinsic mechanical slack that we took advantage of in order to provide corrections. A final area of interest would be to combine our grasp adaptation approach with a model-based approach that can optimally plan an initial grasp and also recover from a loss of contact produced by too strong a perturbation.

ACKNOWLEDGMENT

The research leading to these results has received funding from the Swiss National Science Foundation through the NCRR in Robotics, and the European Community's Seventh Framework Programme FP7/2007-2013 - Challenge 2 - Cognitive Systems, Interaction, Robotics - under grant agreement n^o [231500]-[ROBOSKIN].

REFERENCES

[1] A. M. Dollar, L. P. Jentoft, J. H. Gao, and R. D. Howe, "Contact sensing and grasping performance of compliant hands," *Autonomous Robot*, vol. 28, pp. 65–75, 2010.

[2] O. Kroemer, R. Detry, J. Piater, and J. Peters, "Combining active learning and reactive control for robot grasping," *Robotics and Autonomous Systems*, vol. 58, no. 9, pp. 1105–1116, 2010.

[3] M. Huber and R. Grupen, "Robust finger gaits from closed-loop controllers," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'02)*, Lausanne, Switzerland, 2002.

[4] K. Hsiao, S. Chitta, M. Ciocarlie, and E. G. Jones, "Contact-reactive grasping of objects with partial shape information," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'2010)*, Tapei, Taiwan, 2010, pp. 1228–1235.

[5] T. Takahashi, T. Tsuboi, T. Kishida, Y. Kawanami, S. Shimizu, M. Iribe, T. Fukushima, and M. Fujita, "Adaptive grasping by multi fingered hand with tactile sensor based on robust force and position control," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '08)*, 2008.

[6] A. M. Okamura, N. Smaby, and M. R. Cutkosky, "An overview of dexterous manipulation," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '00)*, 2000.

[7] N. Hogan, "Impedance control: An approach to manipulation: Parts i, ii, and iii," *ASME Journal of Dynamic Systems, Measurement, and Control*, vol. 107, pp. 1–24, 1985.

[8] H. Maekawa, K. Tanie, and K. Komoriya, "Dynamic grasping force control using tactile feedback for grasp of multifingered hand," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '96)*, Minneapolis, Minnesota, USA, 1996.

[9] R. Dillmann, O. Rogalla, M. Ehrenmann, R. Zllner, and M. Bordegoni, "Learning robot behaviour and skills based on human demonstration and advice: The machine learning paradigm," in *Proceedings of the Ninth International Symposium of Robotics Research (ISRR '99)*, Snowbird, Utah, USA, 1999.

[10] A. Lockerd and C. Breazeal, "Tutelage and socially guided robot learning," in *Proceedings of the International Conference on Intelligent Robots and Systems (IROS '04)*, Sendai, Japan, 2004.

[11] S. Calinon and A. Billard, "Incremental learning of gestures by imitation in a humanoid robot," in *Proceedings of the 2nd ACM/IEEE International Conference on Human-Robot Interaction (HRI'07)*, Arlington, Virginia, USA, 2007.

[12] B. Argall, B. Browning, and M. Veloso, "Learning robot motion control with demonstration and advice-operators," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '08)*, Nice, France, 2008.

[13] D. Lee and C. Ott, "Incremental motion primitive learning by physical coaching using impedance control," in *Proceedings of the International Conference on Intelligent Robots and Systems (IROS '10)*, Taipei, Taiwan, 2010.

[14] N. Tsagarakis, G. Metta, G. Sandini, D. Vernon, R. Beira, F. Becchi, L. Righetti, J. Santos-Victor, A. Ijspeert, M. Carrozza, and D. Caldwell, "iCub: The design and realization of an open humanoid platform for cognitive and neuroscience research," *Advanced Robotics*, vol. 21, 2007.

[15] A. Bicchi, "On the closure properties of robotic grasping," *International Journal of Robotic Research*, vol. 14, no. 4, pp. 319–334, 1995.

[16] O. Khatib, "Unified approach for motion and force control of robot manipulators: The operational space formulation," *IEEE Journal of Robotics and Automation*, vol. 3, pp. 43 –53, 1987.

[17] N. S. Pollard, "Closure and quality equivalence for efficient synthesis of grasps from examples," *International Journal of Robotics Research*, vol. 23, pp. 595–613, 2004.

[18] R. D. Howe, "Tactile sensing and control of robotic manipulation," *Journal of Advanced Robotics*, vol. 8, pp. 245–261, 1994.

[19] A. Bicchi, A. Marigo, and D. Prattichizzo, "Dexterity through rolling: Manipulation of unknown objects," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '99)*, 1999.

[20] Z. Li and S. S. Sastry, "Issues in dextrous robot hands," in *Dextrous Robot Hands*, S. T. Venkataraman and T. Iberall, Eds. Springer-Verlag, 1990, pp. 154–186.

[21] R. S. Dahiya, G. Metta, M. Valle, and G. Sandini, "Tactile sensing - from humans to humanoids," *IEEE Transactions on Robotics*, vol. 26, pp. 1–20, 2010.

[22] C. H. Lin, T. W. Erickson, J. A. Fishel, N. Wettels, and G. E. Loeb, "Signal processing and fabrication of a biomimetic tactile sensor array with thermal, force and microvibration modalities," in *Proceedings of the International Conference on Robotics and Biomimetics*, 2009.

[23] H. Yussof, J. Wada, and M. Ohka, "Grasp synthesis based on tactile sensation in robot manipulation of arbitrary located object," in *Proceedings of the IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, 2009.

[24] K. Hsiao, L. Kaelbling, and T. Lozano-Perez, "Task-driven tactile exploration," in *Proceedings of Robotics, Science and Systems (RSS '10)*, 2010.

[25] J. Steffen, R. Haschke, and H. Ritter, "Experience-based and tactile-driven dynamic grasp control," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '07)*, 2007.

[26] R. Tomović, G. A. Bekey, and W. J. Karplus, "A strategy for grasp synthesis with multifingered robot hands," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '87)*, Raleigh, North Carolina, USA, 1987.

[27] A. Petrovskaya, O. Khatib, S. Thrun, and A. Ng, "Bayesian estimation for autonomous object manipulation based on tactile sensors," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '06)*, Orlando, Florida, USA, 2006.

[28] R. Platt, "Learning grasp strategies composed of contact relative motions," in *Proceedings of the IEEE-RAS International Conference on Humanoids Robots*, Tokyo, Japan, 2007.

[29] M. Popović, D. Kraft, L. Bodenhagen, E. Baeski, N. Pugeault, D. Kragic, T. Asfour, and N. Krüger, "A strategy for grasping unknown objects based on co-planarity and colour information," *Robotics and Autonomous Systems*, vol. 58, no. 5, pp. 551–565, 2010.

[30] K. Hsiao, P. Nagneroni, M. Huber, A. Saxena, and A. Ng, "Reactive grasping using optical proximity sensors," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '09)*, Kobe, Japan, 2009.

[31] M. Teichmann and B. Mishra, "Reactive robotics i: Reactive grasping with a modified gripper and multi-fingered hands," *International Journal of Robotics Research*, vol. 19, no. 7, pp. 697–708, 2000.

[32] J. Felip and A. Morales, "Robust sensor-based grasp primitives for a three-finger robot hand," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'09)*, St. Louis, Missouri, USA, 2009.

[33] T. Dorsam, S. . Fatikow, and I. Streit, "Fuzzy-based grasp-force-adaptation for multifingered robot hands," in *Proceedings of the Third IEEE Conference on Fuzzy Systems*, Orlando, Florida, USA, 1994.

[34] L. Ascari, U. Bertocchi, P. Corradi, C. Laschi, and P. Dario, "Bio-inspired grasp control in a robotic hand with massive sensorial input," *Biological Cybernetics*, vol. 100, pp. 109–128, 2009.

[35] R. Jakel, S. R. Schmidt-Rohr, Z. Xue, M. Losch, and R. Dillmann, "Learning of probabilistic grasping strategies using programming by demonstration," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '10)*, 2010.

[36] J. Tegin, J. Wikander, S. Ekvall, D. Kragic, and B. Illev, "Demonstration based learning and control for automatic grasping," *Intelligent Service Robotics*, vol. 3, pp. 23–30, 2008.

[37] M. Muhlig, M. Gienger, and J. J. Steil, "Human-robot interaction for learning and adaptation of object movements," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '09)*, 2009.

[38] R. M. Voyles and P. K. Khosla, "Gesture-based programming: A preliminary demonstration," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '99)*, 1999.

[39] S. Ekvall and D. Kragic, "Interactive grasp learning based on human demonstration," in *Proceedings of IEEE International Conference on Robotics and Automation (ICRA '04)*, 2004.

[40] S. Chernova and M. Veloso, "Learning equivalent action choices from demonstration," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'08)*, Nice, France, 2008.

[41] D. H. Grollman and O. C. Jenkins, "Dogged learning for robots," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'07)*, Rome, Italy, 2007.

[42] B. Argall, E. Sauser, and A. Billard, "Tactile guidance for policy adaptation," *Foundations and Trends in Robotics*, vol. 1, no. 2, 2010.

[43] D. Cohn, Z. Ghahramani, and M. Jordan, "Active learning with statistical models," *Journal of Artificial Intelligence Research*, vol. 4, pp. 129–145, 1996.

[44] S. Calinon, F. Guenter, and A. Billard, "On learning, representing and generalizing a task in a humanoid robot," *IEEE Transactions on Systems, Man and Cybernetics, Part B*, vol. 37, pp. 286–298, 2007.

[45] E. Gribovskaya, S. Khansari Zadeh, and A. Billard, "Learning nonlinear multivariate dynamics of motion in robotic manipulators," *International Journal of Robotics Research*, 2010.

	Range of motion: thumb opposition [deg]			Range of motion: thumb finger [deg]		
$A \rightarrow B$	Ω_2^A	$\Omega_{1:A}^B$	Ω_1^B	Ω_2^A	$\Omega_{1:A}^B$	Ω_1^B
big can→small can	74.8±9.6	92.6±14.0	54.2±13.3	33.7±6.9	32.3±4.4	26.6±4.5
box→big can	85.3±27.0	130.4±0.9	75.8±8.5	34.6±16.3	35.3±6.1	31.2±1.5
small can→box	64.3±3.2	90.6±13.7	61.5±14.5	26.0±3.6	46.1±8.5	31.5±7.9
average value	74.8±18.7	104.5±21.5	63.8±15.3	31.5±11.1	37.9±8.9	29.8±5.8

	Range of motion: index finger [deg]			Range of motion: middle finger [deg]		
$A \rightarrow B$	Ω_2^A	$\Omega_{1:A}^B$	Ω_1^B	Ω_2^A	$\Omega_{1:A}^B$	Ω_1^B
big can→small can	32.0±4.5	29.1±2.6	23.8±0.6	36.2±6.4	34.0±2.9	26.6±5.2
box→big can	37.9±8.7	37.8±11.3	29.4±5.0	36.6±19.4	42.6±12.0	29.0±3.9
small can→box	26.2±1.8	43.4±6.1	26.8±8.6	31.4±9.8	44.8±16.1	29.7±11.6
average value	32.1±7.5	36.7±9.6	26.7±6.2	34.7±13.3	40.5±12.6	28.4±7.8

	Time in force-closure [%]			Contact error [sensor unit]		
$A \rightarrow B$	Ω_2^A	$\Omega_{1:A}^B$	Ω_1^B	Ω_2^A	$\Omega_{1:A}^B$	Ω_1^B
big can→small can	70±9	93±6	96±3	6.4±0.7	5.7±0.8	3.9±1.5
box→big can	82±4	90±3	95±3	4.1±1.2	5.1±1.0	5.2±1.5
small can→box	96±4	94±2	95±1	4.1±1.4	4.1±1.0	4.7±2.2
average value	83±12	92±4	95±3	4.9±1.6	5.0±1.1	4.6±1.8

	Shakiness [deg/s]		
$A \rightarrow B$	Ω_2^A	$\Omega_{1:A}^B$	Ω_1^B
big can→small can	0.09±0.04	0.08±0.04	0.07±0.05
box→big can	0.09±0.05	0.07±0.03	0.07±0.04
small can→box	0.08±0.04	0.06±0.02	0.10±0.05
average value	0.09±0.04	0.07±0.03	0.08±0.05

TABLE II

DETAILED RESULTS OF THE EVALUATION OF POLICY REUSE: DATA ARE GIVEN FOR ALL REUSE COMBINATION TESTED OBJECTS DURING THE DEVELOPMENT OF THE UR MODELS (AVERAGE VALUES ACROSS 3 REPETITIONS OF EACH EXPERIMENT).

[46] A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society Series B*, vol. 39, pp. 1–38, 1977.

[47] A. Schmitz, M. Maggiali, M. Randazzo, L. Natale, and G. Metta, "A prototype fingertip with high spatial resolution pressure sensing for the robot icub," in *Proceedings of IEEE-RAS International Conference on Humanoid Robots*, 2008.

[48] B. Mishra, J. Schwartz, and M. Sharir, "On the existence and synthesis of multifinger positive grips," *Algorithmica, Special issue: Robotics*, vol. 2, pp. 541–558, 1987.

	Range of motion: thumb opposition [deg]					Range of motion: thumb finger [deg]				
	Ω_0	$\tilde{\Omega}_1$	Ω_1	$\tilde{\Omega}_2$	Ω_2	Ω_0	$\tilde{\Omega}_1$	Ω_1	$\tilde{\Omega}_2$	Ω_2
small can	8.7±2.1	33.6±4.2	54.2±13.3	68.0±32.8	94.9±8.1	9.0±1.3	24.2±2.3	26.6±4.5	23.6±8.3	35.1±2.7
big can	9.6±0.8	47.8±10.3	75.8±8.5	76.8±31.8	102.9±1.7	8.0±2.1	17.0±4.9	31.2±1.5	43.4±4.8	42.3±7.0
box	11.5±0.5	40.3±3.2	61.5±14.5	69.5±19.1	88.4±15.3	9.5±4.8	23.3±2.2	31.5±7.9	45.7±16.3	56.4±8.0
ruler	4.9±1.5	12.8±5.8	15.2±2.1	5.3±1.3	18.0±3.9	7.6±0.9	8.9±2.3	12.5±2.5	8.8±7.5	10.9±2.2
average value	8.7±2.8	33.6±14.6	51.7±24.9	54.9±38.0	76.1±35.1	8.5±2.8	18.4±6.9	25.5±9.1	30.4±18.3	36.2±17.4

	Range of motion: index finger [deg]					Range of motion: middle finger [deg]				
	Ω_0	$\tilde{\Omega}_1$	Ω_1	$\tilde{\Omega}_2$	Ω_2	Ω_0	$\tilde{\Omega}_1$	Ω_1	$\tilde{\Omega}_2$	Ω_2
small can	9.5±2.6	21.9±3.1	23.8±0.6	29.9±10.5	33.5±2.6	10.5±1.6	23.8±0.2	26.6±5.2	29.3±7.3	32.0±6.5
big can	10.2±2.4	19.4±2.5	29.4±5.0	41.5±3.6	39.4±2.8	9.0±0.9	19.4±6.9	29.0±3.9	42.9±4.1	44.6±4.7
box	13.2±2.2	17.9±7.4	26.8±8.6	41.9±10.2	52.1±2.5	9.4±4.7	22.8±4.9	29.7±11.6	46.9±14.1	56.0±5.5
ruler	6.9±1.4	10.2±4.9	11.1±0.9	10.4±4.4	12.1±1.6	12.0±0.3	9.3±2.6	23.3±1.8	10.4±6.6	22.2±2.5
average value	9.9±3.1	17.3±6.6	22.8±8.6	30.9±15.0	34.3±14.7	10.2±2.8	18.8±7.3	27.1±7.2	32.4±16.8	38.7±13.7

	Time in force-closure [%]					Contact error [sensor unit]				
	Ω_0	$\tilde{\Omega}_1$	Ω_1	$\tilde{\Omega}_2$	Ω_2	Ω_0	$\tilde{\Omega}_1$	Ω_1	$\tilde{\Omega}_2$	Ω_2
small can	95±4	88±6	96±3	91±4	95±2	5.3±2.0	12.2±8.2	3.9±1.5	15.1±8.4	3.6±1.0
big can	82±9	90±4	95±3	91±4	98±3	5.6±2.7	12.2±1.3	5.2±1.5	10.7±2.2	4.0±1.0
box	75±7	81±10	95±2	90±5	96±0	4.5±3.3	14.9±3.1	4.7±2.2	10.4±4.0	4.1±1.5
ruler	82±5	75±14	87±1	55±24	86±7	5.3±1.4	13.3±5.8	3.7±0.5	13.9±5.0	3.6±1.0
average value	83±10	84±11	94±4	82±20	94±6	5.2±2.5	13.2±5.4	4.4±1.7	12.6±5.8	3.8±1.2

	Shakiness [deg/s]				
	Ω_0	$\tilde{\Omega}_1$	Ω_1	$\tilde{\Omega}_2$	Ω_2
small can	0.06±0.03	0.10±0.05	0.07±0.05	0.06±0.03	0.06±0.03
big can	0.09±0.05	0.07±0.03	0.07±0.04	0.09±0.06	0.08±0.05
box	0.14±0.06	0.12±0.05	0.10±0.05	0.08±0.04	0.08±0.04
ruler	0.09±0.04	0.09±0.05	0.10±0.06	0.12±0.07	0.06±0.05
average value	0.10±0.05	0.10±0.05	0.09±0.05	0.09±0.06	0.07±0.04

TABLE I

DETAILED RESULTS OF THE EVALUATION OF POLICY REFINEMENT: DATA ARE GIVEN FOR ALL TESTED OBJECTS DURING THE DEVELOPMENT OF THE DRR MODELS (AVERAGE VALUES ACROSS 3 REPETITIONS OF EACH EXPERIMENT).