Assessing grasp stability based on learning and haptic data

Yasemin Bekiroglu, Janne Laaksonen, Jimmy Alison Jørgensen, Ville Kyrki and Danica Kragic

***Abstract*— An important ability of a robot that interacts with the environment and manipulates objects, is to deal with the uncertainty in sensory data. Sensory information is necessary to, for example, perform on-line assessment of grasp stability. We present a method for assessing grasp stability based on haptic data and machine learning methods. In particular, we study the effect of different sensory streams to grasp stability. This includes object information such as shape; grasp information such as approach vector; tactile measurements from fingertips and joint configuration of the hand.**

Sensory knowledge affects the success of grasping process both in the planning stage (before a grasp is executed) and during the execution of the grasp (closed-loop on-line control). In this paper, we study both of these aspects. We propose a probabilistic learning framework for assessing grasp stability and demonstrate that knowledge about grasp stability can be inferred using information from tactile sensors. Experiments on both simulated and real data are shown. The results indicate that the idea of exploiting the learning approach is applicable in realistic scenarios which opens a number of interesting venues for the future research.

***Index Terms*— Grasping, Force and Tactile Sensing, Learning and Adaptive Systems.**

I. Introduction

Grasping is an essential skill for a general purpose service robot, working in an industrial or home-like environment. If object parameters such as pose, shape, weight and/or material properties are known, grasp planning using analytical approaches can be employed [1]. In unstructured environments these parameters are uncertain, which presents a great challenge for the current state-of-the-art approaches. Extraction and appropriate modeling of sensor data can alleviate the problem of uncertainty. Many approaches to robotic object grasping exist and most of these have been designed for dealing with known objects. To estimate the shape and pose of an object, visual sensing has been used [2, 3, 4, 5, 6, 7]. However, the accuracy of vision is limited, for example due to imperfect calibration and occlusions. Small errors in object pose are thus common even for known objects and these errors may cause failures in grasping. These failures are commonly difficult to prevent at the grasp execution stage if a hand is not equipped with proper sensors. Tactile and finger force sensors can be used to reduce some problems but are uncommon in practice [8, 9]. Due to uncertainty in the observations, a grasp may fail due to slippage or collision even when all fingers have adequate contact forces and the hand pose with respect to the object is not very different from the planned one.

The main contribution of our work is a new approach that incorporates knowledge of uncertainty in the observations when predicting the stability of a grasp. We show how grasp stability can be assessed based on data extracted both *prior to* and *during* execution. The data contain object information such as shape; grasp information such as approach vector; and online sensory and proprioceptive data including tactile measurements from fingertips and joint configuration of the hand. In real world scenarios the observations acquired from the environment are erroneous and associated with a degree of uncertainty. Our goal is to create a system capable of performing prediction of grasp stability from real world sensory streams. In order for the system to be robust, the uncertainty in the observations needs to be taken into account. Probabilistic methods provide a framework for dealing with uncertainty in a principled manner and will to this end provide the foundation that our system is built upon. Our aim is to model the embodiment specific and inherently complex relationship between grasp stability and the available sensory and proprioceptive information. Our approach is a learning based framework and relies on having a training data-set which can be assumed to sample the domain of possible scenarios well. This poses a challenge as acquiring such data is associated with a significant cost with respect to time and computation. In order to alleviate this problem we use a simulator from which we can generate a large set of synthetic training data in a controlled environment with relative ease. The approach of using synthetic training data is justified by performing inference on real-world examples. Moreover, the generalizability of the grasp stability estimation is experimentally evaluated. The results demonstrate that the stability estimation generalizes relatively well to new objects even with a moderate number of objects used in training. In summary, the paper demonstrates that knowledge about grasp stability can be inferred using information from tactile sensors while grasping an object before the object is further manipulated. This is very useful since, if an unstable grasp is predicted, objects can be regrasped before attempting to further manipulate them.

In the following section, the contributions of our work are discussed in detail in relation to the state-of-the-art work in

Y. Bekiroglu and D. Kragic are with the Centre for Autonomous Systems and Computation Vision and Active Perception Laboratory, School of Computer Science and Communication-CSC, KTH, Stockholm, Sweden. J. Laaksonen and V. Kyrki are with the Department of Information Technology, Lappeenranta University of Technology, Finland. J. A. Jørgensen is with Robotics group, Maersk Mc-Kinney Moller Institute, University of Southern Denmark, Denmark. Janne.Laaksonen,kyrki@lut.fi, yaseminb,danik@csc.kth.se, jimali@mmmi.sdu.dk.

This work was supported by EU through the project CogX, FP7-IP-027657, and GRASP, FP7-IP-215821 and Swedish Foundation for Strategic Research.

the area. This is followed by a presentation of the theoretical framework in Section III and the employed learning methodology. In Section IV the simulator, the database and the real data collection are described. We present the results of experimental evaluation in Section V and conclude our work in Section VI.

II. Contributions and Related work

In robotic object grasping there has been a lot of effort during the past few decades [1]. Grasp stability analysis is a tool often used in grasp planning, where the grasp is planned using grasp quality measures derived from stability analysis. Most of the work on grasp stability assessment relies on analytical methods and focuses on rigid objects, albeit some work has considered the analysis of grasps on deformable objects [10]. Compared to our approach, the analytical methods require exact knowledge of the contacts between the hand and the object to estimate the stability of a grasp.

Most of the grasp planning approaches tested in simulation have the common property of using a strategy that relies on the object shape. Modeling object shape with a number of primitives such as boxes, cylinders, cones, spheres [11, 4], or superquadrics [12] reduces the space of possible grasps. The decision about the suitable grasp is made based on grasp quality measures given contact positions. However, none of these approaches provide a principled way of dealing with uncertainties that arise in dynamic scenarios or the errors inherent to simplification with primitives, which can potentially be solved using tactile feedback. This is also the main objective and contribution of the work presented here.

One of the issues often faced in household scenarios are deformable objects. Planning grasps for these type of objects is not at all as well studied as rigid objects. Examples can be found in literature, such as [13], where the deformation properties of objects are learned and then a suitable grasping force is planned for the associated objects.

To cope with the fact that the exact knowledge of the object and the hand is not available, we employ tactile sensors measuring a range of pressure levels. Tactile sensing has been used for various purposes in prior studies and we focus on the use of tactile sensors in the remaining survey of the related work. There are recent examples which perform grasp generation from visual input and use tactile sensing for closed loop control once in contact with the object. For example, the use of tactile sensors has been proposed to maximize the contact surface for removing a book from a bookshelf [14]. Application of force, visual and tactile feedback to open a sliding door has been proposed in [15]. In our work the main difference is that the tactile sensors are used to assess the stability of a grasp. Thus, rather than using the tactile data for control, we use it in order to reason about grasp stability.

Learning aspects have been considered in the context of grasping mostly for the purpose of understanding human grasping strategies. In [16], it was demonstrated how a robot system can learn grasping by human demonstration using a grasp experience database. The human grasp was recognized with the help of a magnetic tracking system and mapped to the kinematics of the robot hand using a predefined lookup-table. Another approach is to use vision. However, measuring the contact between object and hand accurately is a non-trivial task. The system in [2] learns grasping points by using hand labeled training data in the form of image regions which indicate good grasping regions. A probabilistic decision system is employed on previously unseen objects to determine a good grasping point or region. In [3], vision is used to create grasp affordance hypotheses for objects and refine the grasp affordance hypotheses through grasping. The result is a set of grasps that will produce good grasps on a specific object.

Current learning approaches using tactile sensors are focused on either determining the properties of objects [17, 18, 19] or object recognition [19, 20, 21, 22]. Different properties of objects give valuable information that can be further used in grasp stability analysis. In [17], the pose of the object is determined using a particle filter technique based on the tactile information gained from the contacts between a gripper and the object. Similar work was presented by Hsiao et al. [23] where object localization was performed with knowledge of tactile contacts on specific objects. In [18], the surface type (edge, flat, cylindrical, sphere) of the tactile contact is determined using a neural network. In [19], tactile information extracted from the sensors on a two fingered gripper is used to determine the deformation properties of an object. However, learning or analyzing such object properties through tactile sensors do not answer the question of grasp stability directly compared to the work presented here.

Work on using tactile sensors for recognition of manipulated objects has been reported rather recently. The main approach is to use multiple grasp or manipulation attempts and then learn the object through the haptic input from the manipulations or grasps. Current approaches use either one shot data from the end of the grasps [21, 22] or temporal data collected throughout the grasp or manipulation execution [19, 20]. In [21], a bag-of-words approach is presented which aims to identify objects using touch sensors available on a two fingered gripper. The approach processes tactile images collected by grasping objects at different heights. In [22], a similar approach is taken for a humanoid hand. A more traditional approach to learning is employed with features extracted from tactile images in conjunction with hand joint configurations as input data for the object classifier. In [20] entropy is used to study the performance of various features in order to determine the most useful features in recognizing objects. In this case, a plate covered with tactile sensor was used as the manipulator. However, the object recognition using the recognized good features did not perform as well as in the other presented works. Thus, no attempts have been made on using tactile sensors placed on a robotic hand to predict the stability of a grasp. We have presented the idea of grasp stability prediction using tactile sensors in [24] with some initial results and we extend our work in this paper.

III. Problem formulation and modeling

Determining grasp stability is difficult when factors affecting the stability are uncertain or unknown. We show that with a probabilistic approach it is possible to assess grasp stability using tactile measurements. Mapping from tactile sensor measurements to grasp stability is complex and not injective because of variability in object parameters, grasp and hand types, and the uncertainty inherent in the process. Thus, we consider grasp stability as a probability distribution

$$P(S|H(t), j(t), O, G), \tag{1}$$

where grasp stability, denoted by S, depends on different measured and/or known factors. The factors taken into account in our model are: i) H, force/pressure measurements from tactile sensors; ii) j, joint configuration of the hand; iii) O, object information, e.g., object identity or shape class; and iv) G, information relevant to the grasp, e.g. approach vector and/or hand preshape. Grasp stability, S, is a discrete variable with two possible states: a grasp is either stable or unstable, while the other variables can be discrete or continuous. Our goal is to assess the effect of factors in Eq. (1) to grasp stability by considering different subsets of the variables.

We study the problem using both instantaneous measurements of variables and time-series measurements. With instantaneous measurements, the stability is assessed only from the instant when the robot hand is static and has closed around the object. This approach is referred to as one-shot classification. In contrast, the time-series approach takes into account measurements generated during the whole grasping sequence. The variables H and j are thus represented from time t_0 to t_n where t_0 and t_n represent the start and the end of the grasping sequence respectively. In the case of one-shot classification, we use the measurements once the hand has reached a static configuration, an approach similar to [21]. Thus, we compare the distribution defined by Eq. (1) to one which discards the time series:

$$P(S|H(t_n), j(t_n), O, G). \tag{2}$$

We show that both approaches described by Eq. (1) and Eq. (2) are valid and that grasp stability can be assessed based on them. To study the contribution of object O and grasp knowledge G, we have set up a hierarchy as depicted in Fig. 1. The hierarchy is divided into levels, each with increasing amount of sensory information being available. At the top level of the hierarchy only the information related to the hand itself, H, and j is used. Thus, we estimate

$$P(S|H, j) = \int\int P(S|H, j, O, G)\, p(G|O)\, p(O)\, dO\, dG\ . \tag{3}$$

Considering only sensor information, the overall distribution will be somewhat uninformative — there is significant uncertainty as the same sensor readings can be associated with both stable and unstable grasps for different objects, grasp approach vectors and hand preshapes. Subsequently, when more pieces of information are considered, the estimation of the distribution should be more specific resulting in better discrimination. At the second level, we consider that object

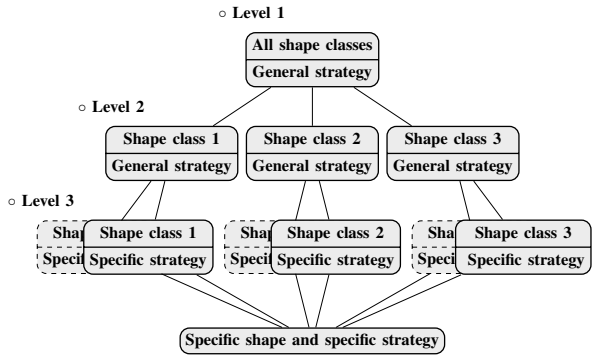

Fig. 1. Hierarchical recognition of grasp stability taking into account different type of sensory knowledge.

shape or object instance are known:

$$P(S|H, j, O) = \int P(S|H, j, O, G)\, p(G)\, dG\ . \tag{4}$$

Finally, at the third level we consider knowledge about the applied grasp, and estimate the stability through $P(S|H, j, O, G)$. Since knowledge of all the variables present in Eq. (1) is assumed, the uncertainty in the stability estimation is expected to decrease.

In the rest of the section, we describe methods for estimating the density functions using a classification approach. Support Vector Machines and AdaBoost are used to model the instantaneous model, according to Eq. (2) while Hidden Markov models are used for the general time series case, according to Eq. (1). Although the probabilistic framework is presented as a method to estimate grasp stability using haptic data, it is also possible to use the proposed framework with other types of sensory information.

A. Feature representation

First, we describe the input features for the classifiers. In this work, a three-fingered Schunk Dextrous Hand (SDH) with seven degrees of freedom and equipped with six two-dimensional Weiss Robotics pressure sensitive tactile pads [25] is used as a demonstration hardware platform. Tactile measurements are recorded from the first contact with the object until a steady state is reached. The whole measurement sequence is denoted by $x_1^i, \ldots, x_{T_i}^i$, where i is the index of the measurement. For one-shot classification, tactile measurements at the steady-state is used and denoted $x_{T_i}^i$. Training data is generated both in simulation and on real hardware and will be presented in Section IV. The notation used in this paper is as follows:

- $D = [o_i], i = 1, \ldots, N$ denotes a data set with N observation sequences.
- $o_i = [x_t^i], t = 1, \ldots, T_i$ is an observation sequence.
- $x_t^i = [M_f^{i,t} j_v^{i,t}], f = 1, \ldots, F, v = 1, \ldots, V$ is the observation at time instant t given the i-th sequence; F

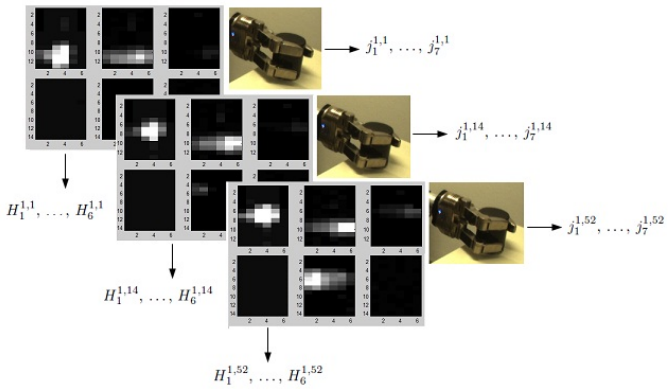

Fig. 2. An example grasping sequence of a cylinder and the corresponding tactile measurements.

is the number of tactile sensors and V is the number of joints of the robot hand.

- $M_f^{i,t}$ includes the moment features extracted from the tactile readings $H_f^{i,t}$ on the sensor f at time instant t given the i-th sequence. Details about the extraction of these features are given later in this section.
- $j_v^{i,t}$ is a joint angle at time instant t given the i-th sequence.

The acquired data thus consists of tactile readings $H_f^{i,t}$ and joint angles of the hand $j_v^{i,t}$. For the Schunk Dextrous Hand, we store $3 \times (14 \times 6)$ readings on proximal and $3 \times (13 \times 6)$ on distal sensors, and seven parameters representing the pose of the hand given the joint angles. Example images from the tactile sensors are shown in Fig. 2. The tactile images in the figure represent a stable grasp of a cylinder.

Tactile data is relatively high dimensional and redundant. Thus, we borrow ideas from image processing and consider the two-dimensional tactile patches as images. Each tactile image is represented using image moments. The general parameterization of image moments is given by

$$m_{p,q} = \sum_z \sum_y z^p y^q f(z,y), \tag{5}$$

where p and q represent the order of the moment, z and y represent the horizontal and vertical position on the tactile patch, and $f(z,y)$ the measured contact. We compute moments up to order two, $(p+q) \in \{0,1,2\}$, for each sensor array separately. These then correspond to the total pressure and the distribution of the pressure in the horizontal and vertical direction. Thus, there are in total six features for each sensor resulting in an observation $x_t^i \in \mathbb{R}^{6F+V}$. Normalizing the feature vector is a common step in machine learning methods. In our case, moment features and finger joint angles are normalized to zero-mean and unit standard deviation. Normalization parameters are calculated from the training data and then used to normalize the testing sequences.

B. One-shot recognition

In this section, we examine the learning of grasp stability based on tactile measurements acquired at the end of a grasping sequence, that is, once the final grasp has been applied to the object. We claim that if successful separation between stable and unstable grasps can be learned from examples, one-shot classification can determine the stability of the grasp from any haptic observation x_t^i measured during a grasp. This information can then be used in grasp control to determine when the robot hand has reached a stable configuration.

In this paper, two types of non-linear classifiers, AdaBoost and Support Vector Machine (SVM), are used in the experiments to demonstrate the ability to learn the stability of the grasps. AdaBoost and SVM were the best performing classifiers in [26]. AdaBoost is a boosting classifier, developed by Freund and Schapire [27], that works with multiple so-called weak learners to form a committee that performs as the classifier. Here, we use AdaBoost implementation from [28].

Support vector machine classification [29, 30] is also suitable for the problem. SVM is a maximum margin classifier, i.e. the classifier fits the decision boundary so that maximum margin between the classes is achieved. This guarantees that the generalization ability between the classes is not lost during the training of the SVM classifier. We use the libSVM implementation presented in [31]. Another critical feature of the SVM for our use is the ability to use non-linear classifiers instead of the original linear hyper-plane classifier. Non-linearity is achieved using different kernels, in this study the radial basis function (RBF)

$$K(x_i, x_j) = e^{-\gamma \|x_i - x_j\|^2}, \quad for \quad \gamma > 0, \tag{6}$$

is used as the kernel for SVM. Moreover, as an extension to the basic two-class SVM, probabilistic outputs for SVM are used to analyze the results given by the SVM. This idea was first presented in [32]. The SVM output $y(\mathbf{x})$ is converted to a probability according to

$$p(t=1|\mathbf{x}) = \sigma(\Gamma y(\mathbf{x}) + \Lambda),\ y(\mathbf{x}) = K(\mathbf{w}, \mathbf{x}) + b\ , \tag{7}$$

where parameters Γ and Λ are estimated using training data, and $\sigma(\cdot)$ is the logistic sigmoid function. This probability is thus related to the earlier general discussion by

$$P(S = stable|H(t), j(t), O, G) = p(t=1|\mathbf{x})\ . \tag{8}$$

C. Temporal recognition using HMMs

Time-series grasp stability assessment is performed using Hidden Markov models (HMMs) [33]. We construct two HMMs: one representing stable and one unstable grasps. Classification of a new grasp sequence is performed by evaluating the likelihood of both models and choosing the one with higher likelihood. For the HMM, we use the classical notation $\lambda = (\pi, A, B)$ where π denotes the initial probability distribution, A is the transition probability matrix

$$A = a_{ij} = P(S_{t+1} = j | S_t = i), i, j = 1 \ldots N, \tag{9}$$

and B defines output (observation) probability distributions $b_j(x) = f_{X_t|S_t}(x|j)$ where $X_t = x$ represents a feature-vector for any given state $S_t = j$. In this work, we evaluate both ergodic (fully connected) and left-to-right HMMs.

The estimation of the HMM model parameters is based on the classical Baum-Welch procedure. The output probability distributions are modeled using Gaussian Mixture Models (GMMs):

$$f_X(x) = \sum_{k=1}^{K} w_k \frac{1}{2\pi^{L/2}\sqrt{|C_k|}} e^{-\frac{1}{2}(x-\mu_k)^T C_k^{-1}(x-\mu_k)}, \quad (10)$$

where $\sum_{k=1}^{K} w_k = 1$, μ_k is the mean vector and C_k is the covariance matrix for the k-th mixture component. The unknown parameters $\theta = (w_k, \mu_k, C_k : k = 1...K)$ are estimated from the training sequences $o = (x_1, ...x_T)$. Initial estimates of the observation densities in Eq. (10) affect the point of convergence of the reestimation formulas. Depending on the structure of the HMM (ergodic vs left-to-right), we use a different initialization method for the parameters of the observation densities. The two initialization procedures are given below:

- For an ergodic HMM, observations are clustered using k-means. Here, k is equal to the number of states in the HMM and each cluster is modeled with a GMM using standard expectation maximization. Initial parameters for the GMMs are found using k-means algorithm.
- For a left-to-right HMM, each observation sequence is divided temporally into equal length subsequences. Then, each GMM is estimated from the collection of corresponding subsequences. Thus, the GMMs represent the temporal evolution of the observations. Initial parameters are found as in the case of an ergodic HMM.

IV. Data collection

For a learning system to achieve good generalization capabilities, relatively large training data is typically required. Generating large datasets on real hardware is time consuming and in robotic grasping generating repeatable experiments is difficult due to the dynamics of the process. However, if suitable models are available, simulation can be used for generation of data for both training the learning system and performance evaluation. In our work, we generate both simulated and real training data as explained below.

A. The simulator

The grasp simulator RobWorkSim, described in [34], is used to generate training data including tactile measurements. The simulator is used in combination with the Open Dynamics Engine (ODE) physics engine and provides support for simulating articulated hands, PD joint controllers, grasp quality measures, camera sensors, range sensors and tactile sensors. The primary motivation for using RobWorkSim over the more widely used GraspIt! [35], is the integrated support for tactile array sensors.

1) Tactile sensor model: The tactile array sensor simulation in RobWorkSim is an experimental model that transforms the point contacts of the ODE to sensor measurements by describing the deformation of the sensor surface given a point force $\mathbf{f}$ applied perpendicular to it. The model was originally described in [36]. The model assumes that the deformation or response is linear with the magnitude of the point force, which is a fair assumption for small forces. Given the deformation function $h(x, y)$ where x and y are specified relative to the center (a, b) of the contact, the total deformation of the surface of an array of rectangular texels with size (A, B) can be found by integrating over the surface of each texel by

$$g_{m,n}(a, b) = \int_{(A-\frac{1}{2})m-a}^{(A+\frac{1}{2})m-a} \int_{(B-\frac{1}{2})n-b}^{(B+\frac{1}{2})n-b} h(x, y)dxdy, \quad (11)$$

where (a, b) is the center point of the contact and (m, n) is the texel index. This surface integration is approximated using the rectangle method. Point force experiments on the real sensors suggested that the deformation decreased with the inverse of the square of the distance from the point force. We use an isotropic function to approximate the deformation of the sensor surface

$$h(x, y) = (\mathbf{f} \cdot \mathbf{n_{texel}})\ max(-\beta + \frac{\alpha}{1 + x^2 + y^2}, 0), \quad (12)$$

where (x, y) is specified relative to (a, b) and $\mathbf{n_{texel}}$ is the normal of the texel on which the point force $\mathbf{f}$ is applied. The parameters (α, β) were found by fitting the model to experimental data extracted from real sensors. Fig. 3 shows a visual comparison between the real and the simulated sensor output where a sharp edge was pressed against both sensors.

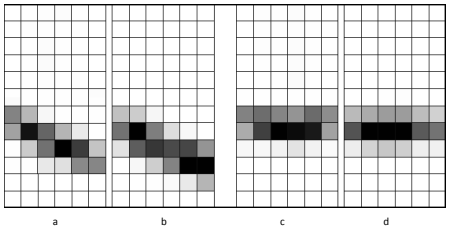

Fig. 3. Measured (a and c) versus simulated (b and d) sensor values. The tactile images were generated by pressing a sharp edge onto the sensor surface.

Assessing grasp quality requires taking properties of the hand (orientation, joint configuration, friction, elasticity, grasping force) and object (shape, mass, friction, contact locations and area, contact force) into account. In the simulated environment these parameters are known. We use a widely known grasp quality measure based on the radius, ϵ, of the largest enclosing ball in the grasp wrench space (GWS). We construct the GWS as proposed in [37] by calculating the convex hull over the set of contact wrenches $\mathbf{w}_{i,j} = [\mathbf{f}_{i,j}^T\ \lambda(\mathbf{d}_i \times \mathbf{f}_{i,j})^T]^T$, where $\mathbf{f}_{i,j}$ belongs to a representative set of forces on the extrema of the friction cone of contact i. $\mathbf{d}_i$ is the vector from the torque origin to contact i and λ weighs the torque quality relative to the force quality.

It is not obvious how to determine λ due to the differences between forces and torques. We therefore calculate force space and torque space independently and use the radius of the largest enclosing ball in each of these to give a 2 dimensional quality value $(\epsilon_f, \epsilon_\tau)$ for each grasp. A third quality measure ϵ_{cmc} based on the distance between the centroid of the contact polygon C and the center of mass CM of the object [38] is used: $\epsilon_{cmc} = ||CM - C||$. This measure captures the same properties as the torque measure, however it is more robust with regard to the point contact output of the simulator. Stable grasps are defined as those for which all three quality values are within a certain threshold. The thresholds have been determined experimentally.

B. Generating training data in simulation

The database includes examples of stable and unstable grasps on different objects. We examine stability starting from the most general case in the hierarchy specified in Fig. 1 and continue by including information about subsequent properties until reaching the most specific case. At the top level of the hierarchy, data is generated on objects with different shapes using approach vectors generated uniformly from a sphere, referred to as a spherical strategy. At the second level, the shape information is given, hence grasps are generated separately per object shape with the spherical strategy. At the third level, the approach vector is formed based on the object shape, namely side or top grasps are applied with more than one preshape. At the bottom level, the preshape is also chosen per object shape and approach vector. Fig. 4 shows examples of objects that are included in the database.

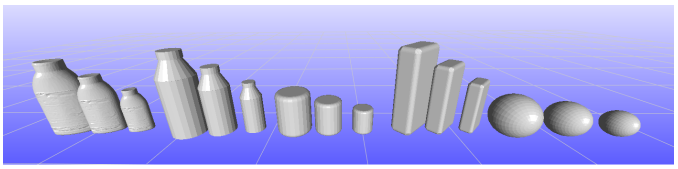

Fig. 4. Objects in simulation were generated in three sizes (75%,100%,125%): Hamburger sauce, Bottle, Cylinder, Box, Sphere.

Each grasping sequence in the database is generated by placing the hand in a specific configuration with respect to the object and then closing the fingers. For the recognition that relates to levels 1 and 2 in the recognition hierarchy (see Fig. 1), a simple spherical grasp strategy with a randomly chosen preshape is used. The spherical grasp strategy generates the approach direction for the hand by sampling the unit sphere around the center of mass of the object. Each sample then consists of a vector pointing toward the center of mass of the object.

The strategy and the preshapes used for level 3 in the recognition hierarchy are shape specific. Therefore strategies where developed for each shape used in the experiments. The hand preshapes for level 3 were generated with finger joint values in the interval $([-90; -70], [-10; 10])^\circ$, where the 7th joint was one of $90^\circ, 60^\circ, 0^\circ$ as shown in Fig. 5.

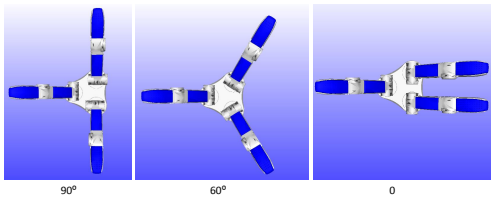

Fig. 5. Hand configuration when the 7'th joint is at 90°, 60° and 0°

The following grasp strategies are applied for the shape primitives:

- Sphere - The approach directions are sampled randomly from the unit sphere with origin in the center of gravity of the object. Both the ball preshape (60°) and the parallel preshape (0°) were used.
- Cylinder - The object is approached either from the top or from the side. When approaching from the top, a ball grasp preshape is used and the approach direction is pointing towards the object center of mass. For side grasps, the approach is sampled with an angle of $0 - 20^\circ$ with respect to the horizontal plane, pointing towards the center of mass of the object. The preshape in the side grasp uses an angle of 0 on joint 7, so that a parallel grasp is obtained.
- Box - The object is approached using a vector lying in the plane defined by the world z-axis and the longest axis of the box and pointing toward the center of gravity. A parallel preshape of the hand is used.

In addition, two natural objects, the hamburger sauce and the bottle (see Fig. 4), used the same strategy as the cylinder. The tactile information and the joint configuration are recorded from simulation at regular time intervals.

In general, the performance of the simulation is largely dependent on the level of detail of the geometries in both hand and objects. In our setup generating a simulated grasp using a modern quad core computer took approximately 2 seconds.

C. Generating training data on a robot

The real world experiments show the feasibility of assessing grasp stability on physical robot platforms. The experiments aim to serve as a proof-of-concept rather than assessing the exact performance rates in different use cases. The experimental evaluation on real data follows the methodology used in simulation such that similar objects and same grasp types are used. The objects are placed such that they are initially not well centered with respect to the hand to assess the ability of the methods to cope with the uncertainty in pose estimation. A few example grasps are shown in Fig. 6. The real data includes side grasps on the objects in Fig. 7 with the preshape shown in Fig. 5 where the 7th joint is 0°. After preshaping, the hand closes the fingers with equal speeds while limiting the maximum torque of each actuator until reaching a static state where the object does not move or a fully closed hand configuration is reached. The latter occurs in the case of an unsuccessful grasp.

Fig. 6. A few examples from the execution of real experiments.

Fig. 7. Objects used in real experiments, with last three deformable.

Tactile readings and corresponding joint configurations were recorded starting from the first contact until a static state is achieved. To generate stable/unstable label for a grasp, the object is lifted and rotated $[-120°, +120°]$ around the approach direction. The grasps where the object is dropped or moved in the hand were labeled as unstable. 100 stable and 100 unstable grasps were generated for each object. Data processing, training and classification followed the same methodology as described for the simulated data.

V. Experiments

We begin the experimental part by describing a simple demonstration scenario to show that the proposed approaches are viable in real applications. As the main experimental contribution, we proceed to study the effect of different types of information for the estimation of grasp stability.

A. Demonstration

The feasibility of the approach is demonstrated in a realistic scenario. The demonstration is included to better show how the proposed methodology can be integrated in a real robotic system. Quantitative evaluation of the methodology is presented after the demonstration.

A vision based system can provide information about the specific objects in the scene and their pose [4, 5, 6] or potential grasping points on the object [39, 7]. In our previous work, we have shown how this can be done for known [4], unknown [5, 6] and familiar objects [39, 7]. However, in the the previous work there were many cases that resulted in unsuccessful grasps. One example using system from [7] is shown in Fig. 8 and more examples are provided in the supplementary material[1].

The scenario that is demonstrated is as follows: Objects of known geometry are placed in the workspace of a robot in a known position similar to [4]. Grasp hypotheses from a planner [40] are applied on the real robot by placing each of the 5 objects (Fig. 9) in a known position. The planner is performing object decomposition for complex objects and plans grasps on the decomposed parts [4]. In our scenario, the planner is configured for a specific preshape. To demonstrate grasping of asymmetric objects in different poses, we place them in four different orientations with respect to the robot. After a suitable grasp is generated by the planner, the hand is moved to a preshape position and the fingers are closed. After a steady state is reached (no change is detected in the tactile sensors), the stability of the grasp is estimated. Finger closing is controlled by executing a constant velocity motion for the finger joints and simultaneously limiting the maximum force by limiting the current for the finger actuators.

Before the system can be operated, a training (calibration) process, required for each individual robotic hand, needs to be completed. The calibration process is described in Algorithm 1. The algorithm is run using the objects in Fig. 9, 114 stable and 114 unstable grasps are generated, including 58 grasps from the white spray bottle and 32 grasps from the pink detergent bottle in Fig. 9. While the calibration algorithm is not tied to a particular classification methodology, in the demonstration the HMM classifier presented in Sec. III-C is shown.

Algorithm 1 Calibration mode.

1: Choose a suitable grasping strategy for object O.
2: **for** $i = 1$ to n **do**
3: Preshape the hand
4: Grasp object O according to the chosen grasping strategy.
5: Record tactile and joint configuration data during the grasp.
6: Manipulate the object O along a predetermined path.
7: Record object motion relative to the hand ΔT.
8: **if** $\Delta T > 0$ **then**
9: Grasp i is unstable.
10: **else**
11: Grasp i is stable.
12: **end if**
13: **end for**
14: Using recorded data from each grasp i, train a classifier C.

The operation mode of the demonstration system is described in Algorithm 2. A grasp is estimated as stable if the probability of a stable grasp exceeds the probability of the grasp being unstable, that is, $P(S = stable) > P(S = unstable)$. The probabilities are estimated using the well-known HMM "forward algorithm" to compute the probability of the observed sequence of measurements, assuming equal prior probabilities for stable and unstable.

Figure 10 shows snapshot images from the operation of the system[2]. The robot is attempting to grasp a bottle by first placing the hand in a preshape position given by the planner mentioned above, as shown in Fig. 10a. Then, the fingers are closed as described above. The closed grasp is shown in Fig. 10b with the corresponding tactile measurements

[1] A supplementary video showing the demonstration is available at http://ieeexplore.ieee.org.

[2] Please see the supplementary video for a more detailed demonstration.

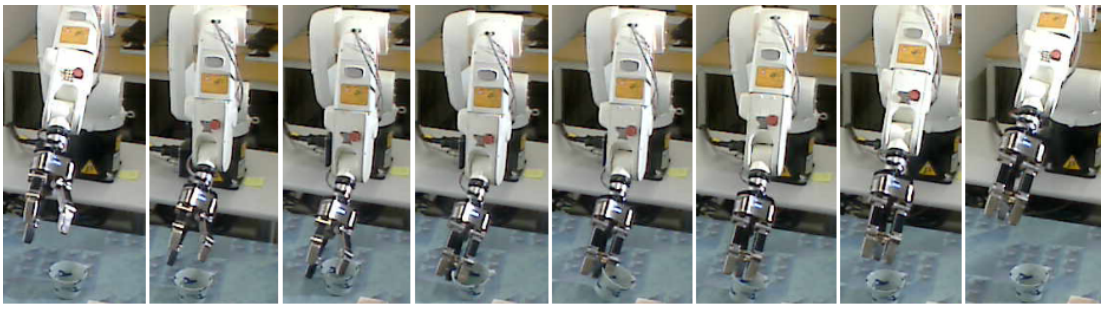

Fig. 8. An example of a failed grasp when only visual input is used. Details about the system are reported in [7].

Algorithm 2 Operation mode.

1: Generate a grasp using our grasp planner.
2: Preshape the hand.
3: Grasp object by closing fingers.
4: Evaluate classifier using sensor data.
5: **if** $P(S = stable) > P(S = unstable)$ **then**
6: Lift object.
7: **else**
8: Go to 1.
9: **end if**

in Fig. 10c. The grasp is predicted to be unstable, with the log-likelihood ratio $\log \frac{P(unstable)}{P(stable)}$ of the two models being $191.1270 > 0$, indicating unstable grasp. Now, in order to demonstrate that the failure was correctly predicted, instead of regrasping, the robot is nevertheless commanded to lift the object. The object drops as shown in Fig. 10d, demonstrating the ability to correctly recognize an unsuccessful grasp. Next, to demonstrate that the stable grasps are also successfully recognized, another grasp generated by the same grasp planner is shown in Fig. 10e. The closed grasp and the corresponding tactile measurements are shown in Figs. 10f and 10g. Based on the measurements, the grasp is predicted to be stable, with the difference across log-likelihoods of the two models being $-537.7687 < 0$, indicating a stable grasp. Lifting and rotating the object around demonstrates this in Fig. 10h, which concludes the demonstration.

Fig. 9. Objects used to generate a dataset for the demonstration.

B. Evaluation of Learning Capability

The experiments are divided according to the hierarchy presented in Section III. The goal is to evaluate the effect of the increasing knowledge on the classification results with both one-shot and temporal classification approaches.

1) Level 1: No constraints: On this level, no constraints are placed on the data used for training the classifiers. In other words, only tactile sensor measurements and the joint configuration are available and the other variables are unknown. The grasps are sampled from a sphere and the hand is oriented towards the object. The data is collected in simulation across multiple object shapes and scales.

2) Level 2: Constraints on object shape: The shape of the object is known, enabling the use of shape specific classifiers. The grasps are randomly sampled from a sphere and the hand is oriented towards the object. The data is collected in simulation.

3) Level 3: Constraints on approach vector, preshape and object shape: On level 3 of the hierarchy, constraints are placed on the approach vector, the grasp preshape and the object shape. The data are collected using a manually chosen approach vector, and the preshape is adjusted to the shape of the object. On this level, the shape is known so that shape specific classifiers can be used. Both simulated data and real data are available at this level.

C. Experimental setup

1) Data: The simulated data used in the experiments consists of five objects with three different grasp configurations applied to them. Three of the objects have primitive shape (box, cylinder, sphere), and two have natural shape (hamburger sauce, bottle). Each object is scaled to three different sizes, 0.75, 1.0, and 1.25 of the original size. For each object/size/grasp combination, 1000 unstable and 1000 stable grasps are randomly chosen from the database described in Sec. IV-B. Thus, each object/grasp dataset consists of 3000 stable and 3000 unstable grasps. When we refer to specific simulated object/grasp combination, terms *side* or *top* are used for grasps generated as side and top grasps, while *sph.* is used for grasps generated uniformly from a sphere around the object (random approach vector). Altogether, there are then 30000 samples for the five objects. We also refer to the root node of the information hierarchy, which contains all samples of primitives shapes, a total of 18000 samples.

The real data collected includes nine objects with 100 unstable and 100 stable grasps for each object. Thus, there are 1800 samples in the real data set. The details of the real data collection are described in Sec. IV-C.

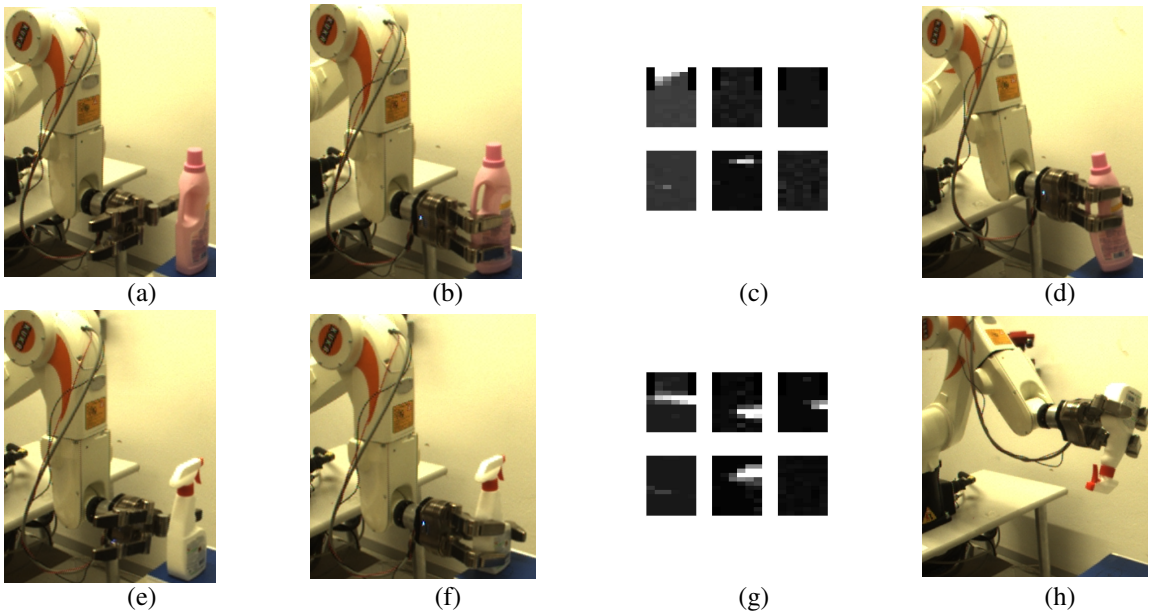

Fig. 10. Operation of the system. First row unsuccessful grasp, second row successful grasp: (a,e) Hand in a preshape position; (b,f) Closed grasp; (c,g) Tactile measurements; (d) The object dropped while lifting; (h) Lifting and rotating the object successfully.

2) One-shot recognition: As mentioned in Section III-B, we utilize the AdaBoost-algorithm in one-shot classification. Due to the formulation of the AdaBoost, a weak learner needs to be chosen. In the experiments, a decision tree with a branching factor of 1 was used as the weak learner, effectively reducing the tree to a series of linear discriminants. The branching factor was determined from series of tests that showed that using branching factor of 1 performed as good or better as larger branching factors on the data described in Sec. IV. 200 iterations of AdaBoost were run to find the final classifier in all experiments. For SVM classifier, $\gamma = 0.03$ and constant C related to the penalty applied to incorrectly classified training samples [29] is set to $C = 0.4$.

All experiments are reported as 10-fold cross validation averages, except where otherwise noted. In each case, the data sets used for training and testing the classifiers are balanced, i.e. the data sets contain equal number of unstable and stable grasps. Image moments are used as the feature representation for the one-shot classifiers. The joint data in addition to the tactile data is also included in the features unless otherwise noted.

3) Temporal recognition: To study if the temporal information improves the recognition performance, two HMMs, one for stable grasps and another for unstable ones, were trained. The stopping criteria for HMM training was a convergence threshold of 10^{-4} with a 10 iteration limit. In order to improve the reliability of the evaluation, both ergodic and left-to-right HMM were evaluated independently. The reason for these multiple experiments is that by evaluating multiple temporal models we aim to understand if the temporal ordering plays part in the modeling. The covariance of the mixture model component distributions was forced to be diagonal.

In the training of the temporal model, the structure of the HMM needs to be chosen in the form of structural parameters, which describe the number of HMM states and the number of mixture model components for each state. These were chosen experimentally such that the HMM was trained using different parameter settings and the setting producing at least lowest equal error rate result (equal number of false positives and negatives) or better performance than that was chosen. The number of states was varied between 2 and 6 while the number of mixture components was between 2 and 5.

Experiments were performed both on simulated and real data. For simulated data randomly chosen 80% of the samples were used for training and the rest 20% for testing. For the real data 10-fold cross validation was used to evaluate the performance and best parameter setting over all folds was chosen. With given parameters, the training time for the HMM varies from less than 20 minutes for the simulated data with thousands of samples to a few minutes for the real data with a few hundreds of samples. Classification of a single sample takes a few seconds.

Image moments were used as features, similar to one-shot learning. However, to reduce the number of parameters in HMM and speed up the training process, principal component analysis (PCA) was applied to the moment and joint measurements separately to reduce the dimensionality of the dataset. The number of principal components was chosen such that at least 99% of the total variance is retained.

D. One-shot recognition

In this section, we present a collection of experiments based on the information hierarchy in Fig. 1 using the AdaBoost classifier. Support vector machine classifier is used with image moments to examine the separability of the grasp stability

at each level by means of log-likelihood histograms. We also study the effect of the joint configuration data on the classification by including or excluding it from the feature vector for the classifier when using real data. Training time for the classifiers is less than five minutes, for the reported amount of samples. Adaboost training time increases linearly with the amount of samples while SVM training time increases quadratically. Classification of a single sample takes less than 10 ms with both of the presented classifiers. SVM classification time increases linearly with the amount of samples used for training.

1) Real data: The experiments begin by showing results using real data. Sampling grasps with a real hand is a slow process and thus the sample size is limited. To study the effect of the amount of samples used for training, we ran a series of tests with variable sample sizes. These tests are shown in Table I. The test shows that for a specific grasp on the cylindrical object, 100 samples are already enough to reach classification performance levels achieved with higher amount of samples, the differences in classification performance above 100 samples are not statistically significant. However, this is the case only when the stable and unstable grasps are distinctive, i.e. we achieve a high rate of correctly classified grasps. In the case of the white bottle data set, where the classification rate is lower, the results show that more than 200 samples could be useful in increasing the classification performance.

TABLE I

ADABOOST CLASSIFICATION RATES (IN PERCENT) ON DATA SETS WITH VARIABLE AMOUNT OF SAMPLES.

Samples	50	100	150	200
Def. cylinder	74.6 %	85.0 %	84.8 %	89.0 %
W. Bottle	64.6 %	68.0 %	68.5 %	75.5 %

Classification results for single object classifiers are presented in columns 2 and 3 of the Table II. Classification rates are shown both with joint configuration data and without it, and the classification rates were computed for image moment feature representations. The main focus in this experiment is to study prediction of the grasp stability on known objects that the system has previously learnt. The average classification rate for known objects is 82.5% including joint data and 81.4% excluding it from the measurements. Thus, the inclusion of joint data seems to benefit the recognition but only to a minor effect. Moreover, the result indicates that at least with known objects the proposed approach seems to have adequate recognition rate for practical usefulness.

We also study how well the trained system can cope with unknown objects, i.e. objects that have not been used to train the system. The results are shown in columns 4 and 5 of the Table II, adjacent to the results with known objects. The results are for a system that has been trained on all the objects except the object for which the classification rate is shown. The average recognition rate is 73.8% with joint data and 72.7% without it. The results show that while the classification rate is lower than with known objects it is still possible to make predictions of the grasp stability on unknown objects to some extent. However, this holds true only when similar grasps are applied on unknown objects as were applied to the objects that the system were trained on. In comparison, including grasps from all objects, including the one being tested, for a single classifier yields a result of 78.6 % correct classification across all the objects in the real object set. This indicates that the variety of objects used in training plays an important role in order to attain good performance, and that the knowledge of object identity is useful but does not seem necessary if the training data includes same or similar objects.

TABLE II

ADABOOST CLASSIFICATION RATES (IN PERCENT) ON KNOWN AND UNKNOWN OBJECTS WITH AND WITHOUT JOINT DATA.

	Known obj.		Unknown obj.	
	w/j	wo/j	w/j	wo/j
Cylinder	88.9 %	90.3	80.4 %	81.9 %
Def. cylinder	91.0 %	89.0 %	76.0 %	76.5 %
Cone	79.5 %	81.0 %	73.0 %	68.0 %
O. Bottle	77.0 %	78.5 %	72.5 %	72.0 %
Shampoo	82.5 %	76.0 %	70.0 %	71.5 %
Pitcher	84.5 %	78.0 %	71.0 %	66.0 %
W. Bottle	76.0 %	73.5 %	75.0 %	76.0 %
B. Bottle	74.0 %	75.0 %	68.5 %	69.0 %
Box	89.0%	91.0 %	78.0 %	73.0 %

Two objects of a primitive shape are included in the real data, a box and a cylinder. Table III shows classification results when the classifier is trained only on one of the primitive objects. The classifier is then asked to classify the grasp stability of grasps made on real-world objects with different shapes. Cross validation was not needed in this case, because the training and test sets are naturally separate. The average classification rate for the cylinder model is 68.0 % and for the box model 66.4 %. These results do not anymore seem adequate for a real system, which again suggests that the variety in the training data is essential.

TABLE III

CLASSIFIER PERFORMANCE (IN PERCENT) WHEN TRAINING WITH A PRIMITIVE OBJECT.

Trained object	**Cylinder**	**Box**
Def. cylinder	76.0 %	73.5 %
Cone	66.0 %	69.5 %
O. Bottle	64.5 %	61.0 %
Shampoo	66.5 %	64.0 %
Pitcher	71.0 %	62.0 %
W. Bottle	73.5 %	69.5 %
B. Bottle	58.5 %	65.0 %

2) Simulated data: In contrast to the real data, in simulation we are able to sample a large number of grasps from different objects and using different grasp strategies. The following classification results were achieved using the simulated data sets described in Section IV. In Table IV, the results are reported for each node in the information hierarchy. The root

node (Level 1) was randomly subsampled to 12000 samples due to computational constraints and has classification rate of 75.3%. The average classification for Level 2 (known object, unknown approach vector) is 76.5% and for Level 3 (known object, known grasp) 77.5%. A trend that increasing knowledge increases classification rate appears, similar to the experiments with real data. However, the trend is significantly weaker compared to the real data. Somewhat surprisingly, the real data classification rates are notably higher when more information is available and the trend is stronger, compared to simulation.

TABLE IV
ADABOOST CLASSIFICATION RATES (IN PERCENT) ACCORDING TO THE INFORMATION HIERARCHY ON SIMULATED DATA.

Level	**Node**	**Classification rate**
Level 1	Root	75.3 %
Level 2	Prim. cylinder sph.	73.5 %
	Prim. box sph.	79.2 %
	Prim. sphere sph.	77.0 %
Level 3	Prim. cylinder side	80.7 %
	Prim. cylinder top	67.6 %
	Prim. box side	83.5 %
	Prim. sphere side	78.5 %

While the primitive shapes used in Table IV are simple shapes, we can use these primitive shapes to train the classifier and then use the classifier to classify grasps sampled from more natural, complex objects. The results are shown in Table V. The table shows results of classifying the natural objects (hamburger sauce, bottle) with different training objects and grasp strategies shown in columns. Comparison results when training the classifier with the natural object and corresponding grasping strategy are shown italic font. The figures in the table show that having data from the correct object has a notable positive effect on the classification rates. This is again a positive argument for the beneficial effect of a variety of training data.

Using the SVM and its ability to output estimates of the prediction certainty, gives us a possibility to examine the performance of the classifier on different data sets in more detail compared to AdaBoost, which supports only the hard decision boundary. This comparison can be seen in Fig. 11. In the figure, log-likelihood ratios, $\log \frac{1-P(S)}{P(S)}$, calculated from the probabilities for stable and unstable samples are shown in histogram form, red for unstable and light blue for stable. The classification errors are shown in filled color, with the filled area indicating the error probability. Fig. 11a-c are from simulated data and Fig. 11d is from the real cylinder grasped with the SDH hand. It is evident from the figure that increasing information makes the distributions for stable and unstable grasps more separate, which was also indicated by the earlier results. Moreover, the figure also supports the finding that classifying the real data seems to be easier than the simulated data. Finally, the figure supports the use of probabilistic approaches for grasp classification, as the ability to measure the uncertainty in classification is important as it can, for example, allow tuning the classification system to give fewer false positives.

E. Recognition based on temporal model

1) Real data: Similar to one-shot classification, we begin by investigating the general performance and the required number of samples for achieving good generalization properties. Table VI shows HMM results corresponding to Table I. The results demonstrate that the performance of HMM classifier does not change much for distinctive grasps such as the ones from the deformable cylinder. While the average classification rates are similar to the one-shot model, the temporal model seems to have better generalization capability in that the classification rate does not decrease significantly with smaller data sets.

TABLE VI
HMM CLASSIFICATION RATES (IN PERCENT) ON DATA SETS WITH VARIABLE AMOUNT OF SAMPLES.

Object	50	100	150	200
Def. cylinder	86.7 %	85.0 %	85.4 %	87.0 %
W. Bottle	78.3 %	82.0 %	74.8 %	75.0 %

Classification results for single object classifiers are presented in Table VII both with joint configuration data (w/j) and without it (wo/j), to study the prediction capabilities on objects the system has previously learnt with the two HMM types (left-to-right: LR, ergodic: ERG). The average classification rate for known objects (with joint data) is 82.4% with LR and 81.7% with ERG which are on a par with the one-shot learning (Table II). Thus, with single object classifiers the inclusion of temporal information did not increase classification performance.

Table VII also includes the results that study how well the trained system can cope with unknown objects, corresponding to Table II for the one-shot learning. The rates not included (marked with a dash) were below the level of chance. The results are similar in the way that the classification rates drop with unknown objects, average rate with joint data being 77.5% for LR and 77.0% for ERG. However, the rate for unknown objects is in most cases high enough such that while the classification rate is lower than with known objects, it is still possible to make useful predictions of the grasp stability on unknown objects. LR seems to outperform ERG slightly in both cases but the difference is not very significant. The reason for the difference is likely to be the simpler structure forced by the LR model, which in turn is likely to prevent overfitting. In comparison, using all data from all objects for a single classifier yields a result of 78.3% for LR model and 76.5% for ERG. It is remarkable that the difference between these and the results without the test object in the training data is less than 1%. Thus, with real data it seems that the generalizability of grasp stability across objects is surprisingly good.

Table VIII shows classification results when the classifier is trained only on one of the primitive objects, corresponding

TABLE V

ADABOOST TRAINING WITH A PRIMITIVE SHAPE AND CLASSIFYING GRASPS SAMPLED FROM A NATURAL OBJECT WITH SIMULATED DATA.

	Prim. cylinder sph.	**Prim. cylinder side**	**Prim. cylinder top**	**Prim. box sph.**	**Prim. box side**	**Prim. sphere sph.**	**Prim. sphere side**	**All classes sph.**
Hamb. sauce	71.5 %	74.0 %	62.9 %	76.8 %	73.6 %	61.4 %	62.7 %	73.4 %
	78.7 %	*83.5 %*	*72.4 %*	*78.7 %*	*82.0 %*	*78.7 %*	*83.5 %*	*78.7 %*
Bottle	68.6 %	77.4 %	56.2 %	72.6 %	76.9 %	59.4 %	66.9 %	69.7 %
	74.7 %	*82.0 %*	*65.2 %*	*74.7 %*	*82.0 %*	*74.7 %*	*82.0 %*	*74.7 %*

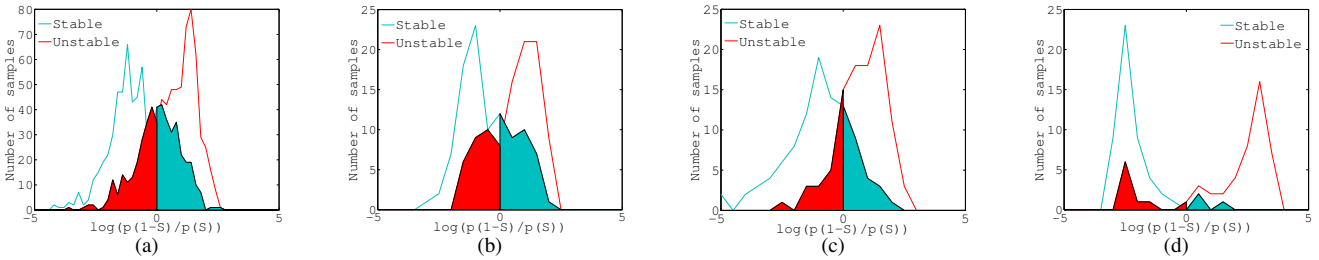

Fig. 11. Likelihood ratios for comparison of separability: (a) Root node, all objects, random grasp vector; (b) Cylinder, random grasp vector; (c) Cylinder side grasp; (d) Real cylinder side grasps.

TABLE VII

HMM CLASSIFICATION RATES (IN PERCENT) ON KNOWN AND UNKNOWN OBJECTS.

	LR, Kn.		ERG, Kn.		LR, Unkn.		ERG, Unkn.	
	w/j	wo/j	w/j	wo/j	w/j	wo/j	w/j	wo/j
Cyl.	90.0	86.5	92.5	82.0	83.0	77.5	81.0	75.0
Def. cyl	87.0	83.5	85.0	83.0	76.0	75.5	76.0	-
Cone	83.0	80.0	81.0	85.0	77.0	73.5	76.0	69.5
O. Bott.	74.0	76.5	75.0	73.5	77.5	77.5	74.5	77.5
Shamp.	81.0	77.5	78.5	77.5	81.0	75.5	79.0	75.0
Pitcher	83.0	81.5	84.0	73.5	72.5	77.5	72.5	65.0
W. Bott.	75.0	69.0	74.0	59.5	77.0	65.0	77.5	-
B. Bott.	78.5	71.0	75.0	66.0	75.5	69.0	75.0	-
Box	90.5	67.0	90.5	68.0	78.5	79.0	81.5	-

TABLE VIII

HMM CLASSIFICATION RATES (IN PERCENT) WHEN TRAINING WITH A PRIMITIVE OBJECT ONLY.

Node	Cylinder		Box	
	LR	ERG	LR	ERG
Def. cylinder	67.0	69.5	74.0	74.5
Cone	66.0	66.0	70.0	76.5
O. Bottle	63.0	60.0	72.0	74.5
Shampoo	61.5	57.5	75.5	77.5
Pitcher	79.5	78.5	-	-
W. Bottle	58.5	50.0	76.5	76.5
B. Bottle	57.0	55.0	73.5	74.5

to one-shot learning results in Table III. The average rate for cylinder primitive is 64.6% for LR and 62.3% for ERG, which are below the results of one-shot recognition. For box primitive, the recognition rate for pitcher was below level of chance and is thus not shown. On average, the rates for box primitive are nevertheless higher than for the cylinder primitive and also higher compared to the one-shot learning. The cause of failure for the single object could not be identified. Altogether, the results are in agreement with those from one-shot learning in that the variety of training data seems important to attain good and stable performance.

2) Simulated data: Using the simulated data, Table IX reports the results for each node in the information hierarchy, corresponding to Table IV for the one-shot learning. For LR model, the average classification for Level 1 (root node, unknown object, unknown approach vector) is 64.9%, 69.9% for Level 2 (known object, unknown approach vector), and for Level 3 (known object, known grasp) 67.5%. The results for ERG are similar. There are two observations to be made. First, these are consistently lower than those with one-shot learning, which is the opposite behavior compared to the real data experiments, indicating that the simulated and real data do not match exactly. Second, the trend that increasing knowledge increases performance is broken for Level 3, although the difference is not very significant. A possible explanation for this is that the stability of top and side grasps is on average more difficult to model with the HMM compared to modeling the stability of a grasp with random approach vector, because it is possible that some of the grasps with random approach vector might be especially easy to recognize correctly.

The classification performance when training with primitive shapes but testing with real-world objects is shown in Table X, corresponding to Table V for the one-shot classification. The classification rates with the correct object are shown in italic for comparison. The results indicate that on average the classification is significantly improved by having the correct object model instead of a general primitive model, again indicating the importance of variety in training data. Moreover, the results are again inferior to one-shot recognition, strengthening the finding that the temporal information is not essential for

TABLE X

HMM TRAINING WITH A PRIMITIVE SHAPE AND CLASSIFYING GRASPS SAMPLED FROM A REAL-WORLD OBJECT WITH SIMULATED DATA.

	cylinder sph.		**cylinder side**		**cylinder top**		**box sph.**		**box side**		**sphere sph.**		**sphere side**		**All sph.**	
	LR	ERG	LR	ERG	LR	ERG	LR	ERG	LR	ERG	LR	ERG	LR	ERG	LR	ERG
Hamb. sauce	61.2	60.8	63.3	60.3	57.8	57.3	59.2	57.3	63.1	61.1	51.6	52.9	65.2	63.2	59.3	59.6
	60.1	*57.2*	*67.5*	*68.0*	*68.1*	*64.8*	*60.1*	*57.2*	*67.5*	*68.0*	*60.1*	*57.2*	*67.5*	*68.0*	*60.1*	*57.2*
Bottle	58.4	58.3	67.6	64.3	63.1	65.4	58.4	54.2	57.2	-	52.4	52.8	62.7	59.6	57.4	58.5
	57.8	*55.6*	*65.8*	*66.8*	*68.8*	*69.1*	*57.8*	*55.6*	*65.8*	*66.8*	*57.8*	*55.6*	*65.8*	*66.8*	*57.8*	*55.6*

TABLE IX

HMM CLASSIFICATION RATES (IN PERCENT) ACCORDING TO THE INFORMATION HIERARCHY ON SIMULATED DATA.

Level	Node	LR	ERG
Level 1	Root	64.9	64.6
Level 2	Prim. cylinder sph.	70.2	70.2
	Prim. box sph.	62.1	59.0
	Prim. sphere sph.	77.4	76.9
Level 3	Prim. cylinder side	69.3	64.3
	Prim. cylinder top	69.5	69.3
	Prim. box side	68.6	69.0
	Prim. sphere side	62.8	63.2

recognition with the available simulated data. To conclude, the real-world cases seem to contain dynamic phenomena which can be modeled better using a temporal model.

VI. CONCLUSION AND FUTURE WORK

Uncertainty is inherent to the activities robots perform in unstructured environments. Probabilistic techniques have demonstrated the strength of coping with the uncertainty in robot planning, decision making, localization and navigation. In the area of robot grasping, there have been very few examples of solving problems such as assessing grasp stability by taking uncertainty into consideration.

In the present work, it was shown how grasp stability can be assessed based on uncertain sensory data using machine learning techniques. Our learning framework takes into account object shape, approach vector, tactile data and joint configuration of the hand. We have used a simulated environment to generate training sequences, including the simulation of the sensors. The methods were evaluated both on simulated and real data using a three-fingered robot hand. Our work demonstrates how grasp stability can be inferred using information from tactile sensors while grasping an object before the object is further manipulated or during the manipulation step. We have implemented and evaluated both one-shot and temporal learning techniques. One focus of the experiments was to study prediction capabilities of the proposed methods for known objects. We have also studied how the system can cope with unknown objects, i.e. objects that have not been used in the training step. The results show that while the classification rate is lower than with known objects it is still possible to make useful predictions of the grasp stability on unknown objects. In summary, the experimental results show that tactile measurements allow assessment of grasp stability. The aim of the paper was not a perfect discrimination between successful and unsuccessful grasps but rather a measure of certainty of grasp stability. This also means that a system may be built to reject some stable grasps while having fewer unstable grasps classified as stable ones. Experiments showed that using sequential data to evaluate grasp stability appears to be beneficial during dynamic grasp execution.

Our current work proceeds in several directions. First, we are in the process of integrating the presented system with a vision based pose estimation system and grasp planning. Second, we are implementing a grasping system based on the proposed ideas for local control of grasps and corrective movements. In both cases, the aim is to demonstrate a robust object grasping and manipulation system for both known and unknown objects based on visual and tactile sensing. Finally, we are developing a more elaborate probabilistic framework in which we study the joint probability of object-relative gripper configurations, tactile perceptions, and grasping feasibility. Here, we are developing a kernel-logistic-regression model of pose- and touch-conditional grasp success probability. The goal is to show how a learning framework can be used for grasp transfer, i.e. if the robot has learnt how to grasp one type or category of objects, to use this knowledge to grasp a new object.

REFERENCES

[1] D. Prattichizzo and J. C. Trinkle, "Grasping," in *Springer Handbook of Robotics*, B. Siciliano and O. Khatib, Eds. Springer, 2008, ch. 28.

[2] A. Saxena, J. Driemeyer, and A. Y. Ng, "Robotic grasping of novel objects using vision," *International Journal of Robotics Research*, vol. 27, no. 2, pp. 157–173, 2008.

[3] R. Detry, E. Baseski, M. Popovic, Y. Touati, N. Krueger, O. Kroemer, J. Peters, and J. Piater, "Learning continuous grasp affordances by sensorimotor exploration," in *From Motor Learning To Interaction Learning in Robots*, 1st ed., O. Sigaud and J. Peters, Eds. Berlin, Germany: Springer-Verlag, 2010.

[4] K. Huebner, K. Welke, M. Przybylski, N. Vahrenkamp, T. Asfour, D. Kragic, and R. Dillmann, "Grasping known objects with humanoid robots: A box-based approach," in *14th International Conference on Advanced Robotics*, Munich, Germany, June 2009.

[5] B. Rasolzadeh, M. Bjorkman, K. Huebner, and D. Kragic, "An active vision system for detecting, fixating and manipulating objects in real world," *International Journal of Robotics Research*, vol. 29, no. 2-3, pp. 133–154, 2010.

[6] M. Popovic, D. Kraft, L. Bodenhagen, E. Baseski, N. Pugeault, D. Kragic, T. Asfour, and N. Kruger, "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.

[7] J. Bohg and D. Kragic, "Learning grasping points with shape context," *Robotics and Autonomous Systems*, vol. 59, no. 4, pp. 362–377, 2010.

[8] M. Shimojo, T. Araki, A. Ming, and M. Ishikawa, "A high-speed mesh of tactile sensors fitting arbitrary surfaces," *IEEE SENSORS JOURNAL*, vol. 10, no. 4, pp. 822–830, 2010.

[9] M. Higashimori, M. Kaneko, A. Namiki, and M. Ishikawa, "Design of the 100g capturing robot based on dynamic preshaping," *International Journal of Robotics Research*, vol. 24, no. 9, pp. 743–753, 2005.

[10] H. Wakamatsu, S. Hirai, and K. Iwata, "Static analysis of deformable object grasping based on bounded force closure," in *International Conference on Robotics and Automation*, Minneapolis, USA, April 1996, pp. 3324–3329.

[11] A. T. Miller, S. Knoop, H. I. Christensen, and P. K. Allen, "Automatic Grasp Planning Using Shape Primitives," in *IEEE International Conference on Robotics and Automation*, 2003, pp. 1824–1829.

[12] C. Goldfeder, P. K. Allen, C. Lackner, and R. Pelossof, "Grasp Planning Via Decomposition Trees," in *IEEE International Conference on Robotics and Automation*, 2007, pp. 4679–4684.

[13] A. M. Howard and G. A. Bekey, "Intelligent learning for deformable object manipulation," *Autonomous Robots*, vol. 9, no. 1, pp. 51–58, 2000.

[14] A. Morales, M. Prats, P. Sanz, and A. P. Pobil, "An experiment in the use of manipulation primitives and tactile perception for reactive grasping," in *Robotics: Science and Systems, Workshop on Robot Manipulation: Sensing and Adapting to the Real World*, Atlanta, USA, 2007.

[15] M. Prats, P. Sanz, and A. del Pobil, "Vision-tactile-force integration and robot physical interaction," in *IEEE International Conference on Robotics and Automation*, Kobe, Japan, 2009, pp. 3975–3980.

[16] S. Ekvall and D. Kragic, "Learning and Evaluation of the Approach Vector for Automatic Grasp Generation and Planning," in *IEEE International Conference on Robotics and Automation*, 2007, pp. 4715–4720.

[17] A. Petrovskaya, O. Khatib, S. Thrun, and A. Y. Ng, "Bayesian estimation for autonomous object manipulation based on tactile sensors," in *IEEE International Conference on Robotics and Automation*, Orlando, FL, USA, May 2006, pp. 707–714.

[18] A. Jiméneza, A. Soembagijob, D. Reynaertsb, H. V. Brusselb, R. Ceresa, and J. Ponsa., "Featureless classification of tactile contacts in a gripper using neural networks," *Sensors and Actuators A: Physical*, vol. 62, no. 1–3, pp. 488–491, 1997.

[19] S. Chitta, M. Piccoli, and J. Sturm, "Tactile object class and internal state recognition for mobile manipulation," in *IEEE International Conference on Robotics and Automation*, Anchorage, AK, USA, May 2010, pp. 2342–2348.

[20] M. Schöpfer, M. Pardowitz, and H. J. Ritter, "Using entropy for dimension reduction of tactile data," in *14th International Conference on Advanced Robotics*, Munich, Germany, June 2009.

[21] A. Schneider, J. Sturm, C. Stachniss, M. Reisert, H. Burkhardt, and W. Burgard, "Object identification with tactile sensors using bag-of-features," in *IEEE/RSJ international conference on Intelligent robots and systems*, St. Louis, MO, USA, 2009, pp. 243–248.

[22] N. Gorges, S. E. Navarro, D. Göger, and H. Wörn, "Haptic object recognition using passive joints and haptic key features," in *IEEE International Conference on Robotics and Automation*, Anchorage, AK, USA, May 2010, pp. 2349–2355.

[23] K. Hsiao, L. Kaelbling, and T. Lozano-Perez, "Task-driven tactile exploration," in *Robotics: Science and Systems*, Zaragoza, Spain, June 2010.

[24] Y. Bekiroglu, J. Laaksonen, J. A. Jorgensen, and V. Kyrki, "Learning grasp stability based on haptic data," in *Robotics: Science and Systems Workshop on Representations for Object Grasping and Manipulation in Single and Dual Arm Tasks*, Zaragoza, Spain, June 2010.

[25] "Weiss robotics tactile sensor." [Online]. Available: http://www.weiss-robotics.de/en.html

[26] J. Laaksonen, V. Kyrki, and D. Kragic, "Evaluation of feature representation and machine learning methods in grasp stability learning," in *10th IEEE-RAS International Conference on Humanoid Robots*, 2010, pp. 112–117.

[27] Y. Freund and R. E. Shapire, "Experiments with a new boosting algorithm," in *Thirteenth Internation Conference on Machine Learning*. Morgan Kaufmann, 1996, pp. 148–156.

[28] A. Vezhnevets, *GML AdaBoost Matlab Toolbox*, 2006, available at http://graphics.cs.msu.ru/ru/science/research/machinelearning/ adaboost-toolbox.

[29] C. Cortes and V. Vapnik, "Support vector networks," *Machine Learning*, vol. 20, pp. 273–297, 1995.

[30] V. Vapnik, *The Nature of Statistical Learning Theory*. New York: Springer-Verlag, 1995.

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

[32] J. C. Platt, "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods," in *Advances in Large Margin Classifiers*. MIT Press, 1999, pp. 61–74.

[33] L. R. Rabiner, "A tutorial on Hidden Markov Models and selected applications in speech recognition," in *Proceedings of the IEEE*, vol. 77, no. 2, 1989, pp. 257–286.

[34] J. Jorgensen, L. Ellekilde, and H. Petersen, "RobWorkSim - an open simulator for sensor based grasping," in *41st International Symposium on Robotics (ISR 2010) and ROBOTIK 2010*, Munich, Germany, June 2010.

[35] A. T. Miller and P. K. Allen, "Graspit! A Versatile Simulator for Robotic Grasping," *IEEE Robotics and Automation Magazine*, vol. 11, no. 4, pp. 110–122, 2004.

[36] J. A. Jorgensen and H. G. Petersen, "Usage of simulations to plan stable grasping of unknown objects with a 3-fingered schunk hand," in *IROS'08 Workshop on Robot Simulators*, Nice, France, September 2008.

[37] C. Ferrari and J. Canny, "Planning optimal grasps," in *IEEE International Conference on Robotics and Automation*, Nice, France, May 1992, pp. 2290–2295.

[38] R. Suárez, M. Roa, and J. Cornella, "Grasp quality measures," Technical University of Catalonia, Tech. Rep. IOC-DT-P-2006-10, 2006.

[39] J. Bohg and D. Kragic, "Grasping familiar objects using shape context," in *14th International Conference on Advanced Robotics*, Munich, Germany, June 2009.

[40] K. Huebner, "BADGr - A Toolbox for Box-based Approximation, Decomposition and GRasping," in *IROS 2010 Workshop on Grasp Planning and Task Learning by Imitation*, Taipei, Taiwan, October 2010.