Robotics and Autonomous Systems 60 (2012) 411–423

Contents lists available at SciVerse ScienceDirect

Robotics and Autonomous Systems

journal homepage: www.elsevier.com/locate/robot

Templates for pre-grasp sliding interactions

Daniel Kappler [b,*], Lillian Y. Chang [a,c,**], Nancy S. Pollard [a], Tamim Asfour [b], Rüdiger Dillmann [b]

[a] Robotics Institute, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, United States
[b] Institute for Anthropomatics, Karlsruhe Institute of Technology, Karlsruhe, Germany
[c] Intel Corporation, Intel Science and Technology Center, University of Washington, Seattle, WA, United States

ARTICLE INFO

Article history:
Available online 7 September 2011

Keywords:
Pre-grasp interaction
Object manipulation
Humanoid robotics
Pushing
Sliding

ABSTRACT

In manipulation tasks that require object acquisition, pre-grasp interaction such as sliding adjusts the object in the environment before grasping. This change in object placement can improve grasping success by making desired grasps reachable. However, the additional sliding action prior to grasping introduces more complexity to the motion planning process, since the hand pose relative to the object does not need to remain fixed during the pre-grasp interaction. Furthermore, anthropomorphic hands in humanoid robots have several degrees of freedom that could be utilized to improve the object interaction beyond a fixed grasp shape. We present a framework for synthesizing pre-grasp interactions for high-dimensional anthropomorphic manipulators. The motion planning is tractable because information from pre-grasp manipulation examples reduces the search space to promising hand poses and shapes. In particular, we show the value of organizing the example data according to object category templates. The template information focuses the search based on the object features, resulting in increased success of adapting a template pose and decreased planning time.

1. Introduction

In service tasks involving object fetching or transport, an autonomous manipulator must acquire the object before delivery. When the desired contact surfaces for grasping are within reach, a direct reach-to-grasp motion is sufficient to achieve object acquisition. However, in unstructured environments such as the home or office, object placement may change from day-to-day and might not always be convenient for reaching the desired grasp. In these scenarios, *pre-grasp interaction* with a movable object can adjust the object placement to improve the reachability of good grasps. Human examples of pre-grasp interaction include sliding flat objects such as a credit card to a table edge to grasp it, pushing a heavy box near the body mass center for easier lifting, or rotating a handled object such as a water pitcher.

In this paper, we present a method for synthesizing pre-grasp sliding interactions for an anthropomorphic manipulator (Fig. 1). Pregrasp interactions are synthesized by the proposed framework either completely in simulation and then executed on the robot, or integrated into the robot execution loop. Both versions are evaluated in the experiments. Many service robots have a humanoid form designed to perform manipulation tasks with human-like motions [1–3]. The multi-fingered hands of these robots have several degrees of freedom (DoF) for achieving a wide range of hand shapes to accommodate different object geometries. However, the manipulator's kinematic freedoms as well as additional object motion introduce more complexity to the planning process for pre-grasp interactions.

To make the planning tractable, our framework makes use of human pre-grasp manipulation examples to narrow the search to promising hand poses for the pushing interaction. We have observed that in human pre-grasp interaction of objects on a tabletop, the manipulation tended to include pre-grasp sliding toward the body if the object is out of reach or inconvenient to grasp. These sliding manipulations also tended to exhibit similar hand shapes and poses for objects of similar shape and weight. These patterns form the basis of our framework, where pre-grasp interaction examples are organized into templates based on object categories.

In our framework, example hand preshapes for pre-grasp sliding are stored in the example database according to the object category to retain the context information of the template. An object category for a target object is obtained based on the object appearance. The information stored for one object template includes the starting hand preshape consisting of a hand pose relative to the object and the finger joint configurations, a set

* Corresponding author.
** Corresponding author at: Robotics Institute, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, United States.
E-mail addresses: daniel.kappler@student.kit.edu (D. Kappler), lillianc@cs.cmu.edu (L.Y. Chang), nsp@cs.cmu.edu (N.S. Pollard), asfour@kit.edu (T. Asfour), dillmann@kit.edu (R. Dillmann).

0921-8890/$ – see front matter
doi:10.1016/j.robot.2011.07.015

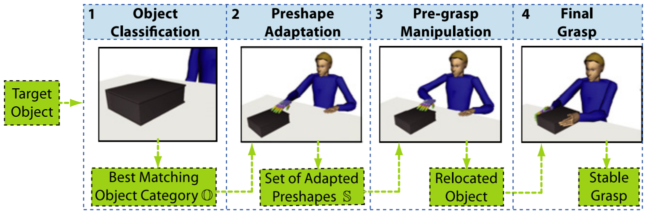

Fig. 1. Overview of the pre-grasp strategy, where an object is slid on the table before the final grasp.

of hand preshapes associated with the final grasp to achieve object acquisition, and the constraints on the related pre-grasp interaction strategies. Constraints on the pre-grasp interaction strategies include context information such as final goal regions for the object relocation, as well as constraints on the object motion or robot joint motion. Although the data structures proposed in this paper are designed to work with different pre-grasp strategies and grasp planners, for evaluation we implemented sliding as the type of pre-grasp manipulation. For evaluation, 3D triangular meshes of the objects are currently needed for planning, and the framework assumes that the target object can be slid over the surface as long as finger contact is enforced, which is a reasonable assumption as demonstrated by the experiments described in Section 6.5.

These preshape templates enable efficient automatic generation of hand poses for pre-grasp object interaction, which ultimately improves the success of grasping for object acquisition. In particular, we show the value of organizing the example data according to object category templates. The template information further focuses the search based on the object features, increasing the success of adapting a template pose and decreasing the planning time.

In the following, we first review previous work on grasping templates and manipulation planning related to the four stages of our proposed framework. We then describe the collection of human pre-grasp interaction examples. The remainder of the paper presents our template-based framework for planning pre-grasp sliding motions, our experimental results and validation tests, and a discussion of future steps to generalize this framework for an even broader set of manipulation actions.

2. Related work

Our framework for planning pre-grasp interactions is composed of four main phases (Fig. 1): (1) object classification to determine the object category, (2) starting pose generation, adaptation and evaluation of candidate hand preshapes from the initial template, (3) simulation of the pre-grasp interaction that adjusts object placement, and (4) planning the final grasp which acquires the object. We describe here the previous work related to the different components of our framework.

The first phase of object classification determines a category, e.g. box or cylinder, based on sub-symbolic object features such as shape and weight. Approaches for solving this problem include, for example, neural networks [4] or support vector machines [5]. For the usage in our proposed method it is possible to use offline, online, supervised or unsupervised learning. This category narrows the set of candidate hand preshapes to those associated with similar objects. Previous investigation of human grasps has explored how a set or taxonomy of prototypical hand shapes can describe the space of hand configurations for grasping tasks [6–8]. In imitation learning, several researchers have presented methods of learning the classification of a human demonstration of a grasp, which can then be mapped to a robot hand configuration [9–13]. In our work, the classification is on object features rather than the hand shape itself. In this regard, it is most similar to the learning methods in [14–16] where features of an object's component sub-shapes are used to learn which sub-shape is a handle or suitable handshapes for grasping the sub-shape. Our method does not use object decomposition for determining grasp contacts. The contact surfaces on the object are determined in later phases of our framework based on the preshape examples associated with each object category.

Both the second phase of preshape adaptation and the fourth phase of final grasp planning involve modification of template hand shapes to fit a new object geometry. A method to refine a prototype hand shape has been developed [17] to fit a static hand preshape to the surface of a new object. Our method for adapting hand configurations for pre-grasp interactions is similar in the adjustment of the finger joints to achieve more contact with the object. In our framework, though, the preshape adaptation is evaluated for non-grasping (non-prehensile) hand poses by how well the contact forces contribute to the desired object adjustment in pre-grasp interaction. Other methods of grasp template refinement to new object shapes have also been developed for synthesizing new contact points on the object surface [18,19], from image analysis instead of 3-D geometry [20], and using grasp synergy subspaces [21].

The third phase of planning the object interaction motion is related to the field of manipulation planning. There are several strategies of pre-grasp interaction manipulation for adjusting the object on the support surface. Our framework is intentionally designed to accommodate multiple modes of interaction, and this paper primarily discusses pre-grasp interaction by planar pushing as a widely applicable action. The simplest form of pushing is single-freedom reorientation in the plane, which has been used in pre-grasp rotation to grasp hard-to-reach object handles [22–24]. Planning techniques for more general pushing and sliding actions using non-prehensile manipulation are discussed in [25–27]. Recent work [28] uses a short push or push-grasp as a type of pre-grasp action primitive for bringing objects into the hand after a reaching motion. Toppling as well as tumbling presented by Lynch et al. [29] are also possible pre-grasp manipulations. For humanoid robots, multi-modal interaction combining locomotion and object pushing has been developed by [30]. A whole-body manipulation strategy for pivoting large, heavy objects has been presented by [31] as the primary manipulation task that avoids completely grasping or lifting the object.

Grasp synthesis without template context has also been investigated from several aspects. Berenson et al. [32] proposed a method to precompute a set of possible grasps offline for later online evaluation against environmental constraints. Przybylski et al. suggested in [33] to use the medial axis to reduce the number of possible grasp candidates. Finding model-based analytical grasps is discussed by Bicchi and Kumar in [34]. For evaluating the stability of a possible lifting grasp, Ferrari and Canny [35] proposed the force closure metric, which measures the force exerted on the object at the contact points. More recent research about grasp quality is presented by Miller and Allen [36].

Our presented framework also builds upon multiple concepts of motion planning for synthesizing arm actions. Our templates for pre-grasp interaction hand shapes store context information about the hand shape and the hand pose relative to the object. The examples do not include arm posture configurations, which would be specific to a particular base placement of the robot relative to the desired hand pose. Thus, it is necessary to not only synthesize hand shapes but also collision-free arm postures to reach the desired hand pose. Many humanoid robots have at least 7-Degrees of Freedom (DoF) arms which leads to redundant inverse kinematics problems to achieve hand poses. This is an extensively studied topic in the computer graphics and robotics community [37–39]. These approaches enable computation of manipulator configurations for grasping, sliding and rotating manipulations. For a complete motion, a path between the initial pose and the calculated one has to be planned. Additional constraints for such a path are obstacle avoidance and joint limitations. Latombe [40] discusses traditional approaches to this problem, and many recent works on motion planning have presented variants of the Rapidly-exploring Random Tree (RRT) method proposed by Lavalle [41].

Current methods [15,32,33] for planning object fetching search direct grasp solutions based on a known object pose. Our pre-grasp strategy augments such methods by a preliminary step which reconfigures the object pose through suitable pre-grasp manipulation actions. This strategy not only results in higher success rates for finding stable grasp solutions, but it also increases the final grasp stability for grasping. Also, even when a direct grasp solution is available, our approach enables faster online planning based on the focused search from context knowledge.

3. Human examples of pre-grasp interaction

In many scientific fields, successful techniques have been inspired by studying solutions provided by nature. Pre-grasp interaction is a human-inspired manipulation strategy that has been observed in many typical household activities in our previous study [42]. In addition, our detailed study [43] of human performance of a specific type of interaction, pre-grasp rotation, has inspired the development of similar techniques for object re-orientation on robot manipulators [22,24].

In this work, we use examples from human actions for more general interactions during the grasping process, not just pre-grasp rotation. Our initial survey provided insight into possible underlying patterns for pre-grasp interaction with objects on a cluttered surface. Our follow-up observations instrumented the objects and people to capture specific examples of their hand motions relative to the object. These examples form the basis of our database of preshapes templates for the pre-grasp interaction.

Both studies of human pre-grasp interaction were conducted with participants who provided their voluntary consent, in accordance with the Carnegie Mellon Institutional Review Board policies.

3.1. Video survey

We first conducted an informal video survey to investigate different human strategies and possible object-based patterns. For each participant various objects such as a stapler, books, and CDs were randomly placed on a table. The participants were instructed to remove the objects from the table and place them on a nearby chair. They sat at the table while performing the task and were permitted to use only one hand. The major strategy we observed was that people tend to perform pre-grasp manipulation, especially sliding the object on the table, to pick up the object. Furthermore, they used similar hand poses for pre-grasp manipulation when grasping objects of similar shapes. Additionally we noticed that the participants slid the object to different regions depending on object appearance and final grasp type which supports separation of the data into the proposed data structures *object category*, *preshape*, and *pre-grasp manipulation*.

3.2. Motion apture

In a second observation of human examples, we ran a motion capture experiment to record pre-grasp manipulation motion. The purpose of this experiment was to collect a small number of real examples of the hand shapes used for pre-grasp pushing, rather than formally measure and completely describe the human motions. The experiments resulted in two sets of data for:

1. the hand positions relative to the object and
2. the hand preshape configuration,

both just before contact with the object. We expected that both the hand position and preshape would change for different object categories and different environmental conditions. Here, environmental conditions refers to the object poses relative to the human pose.

The setup consisted of a table with randomly-placed objects, and the participants were instructed to move all objects to another table. The setup required participants to walk a few meters between the tables, which was intended to implicitly force the use of stable object grasps for acquisition. Unlike the video survey, there was no constraint for one-handed grasps in this experiment, because we preferred to capture the natural grasping behavior as much as possible.

We captured examples of hand positions and configurations from four adult participants. These data examples were recorded using a motion capture system (Vicon), enabling full body, hand and object tracking using 3D point clouds. The recorded data was used to generate our preshapes based on manually-extracted motion data for which we manually fitted the kinematic hand model and object model. Goal regions, in which grasps of a certain kind have a high probability for success, were generated based on the video survey and the motion capture observations.

In some cases, the participants held multiple objects in one hand while acquiring the next object with the other hand. We hypothesize this may be an effort to reduce the number of walks to the second table. All participants used pre-grasp manipulation to slide or reorient a few objects, and three used it for most objects. The grasp shapes and positioning on the object were similar between different participants.

4. Pre-grasp strategy

We present a data-driven strategy which is designed to automatically perform pre-grasp manipulation actions to fetch an object from a surface. That is, given an object in an initial pose on a surface, our method plans a solution for a robot in simulation to adjust the object to a final pose, within a distinct final region, using a suitable pre-grasp manipulation strategy. Fig. 2 illustrates in detail the framework introduced in Fig. 1. This framework can be integrated into a real robot platform performing open-loop execution of the synthesized plan as demonstrated in Section 6.5. This data-driven approach is based on context knowledge consisting of look-up structures for object categories, hand configurations and poses, discrete actions, constraints, goal regions, and grasp-types for which we describe the general concepts and data structures in the following sections. Implementation details for the main phases used for evaluation can be found in Section 5.

4.1. Representations

At the high level, the context knowledge is organized by *object categories* $\mathbb{O}$ which are selected by a classifier using features such as the appearance of the current target object. For each object category, there are multiple entries for different pre-grasp manipulation contexts which allow successful pre-grasp manipulation for a given object within this category with a high probability. The context per object category consists of two parts: first, a set of *preshapes* $\mathbb{S}$ providing knowledge to efficiently find hand configurations, and second, a set of *pre-grasp manipulations* $\mathbb{M}$ to relocate the target object prior to grasping.

$$\mathbb{O} = (\mathbb{S}, \mathbb{M}). \quad (1)$$

The next sections describe the *preshape* $\mathbb{S}$ and *pre-grasp manipulation* $\mathbb{M}$ data structures.

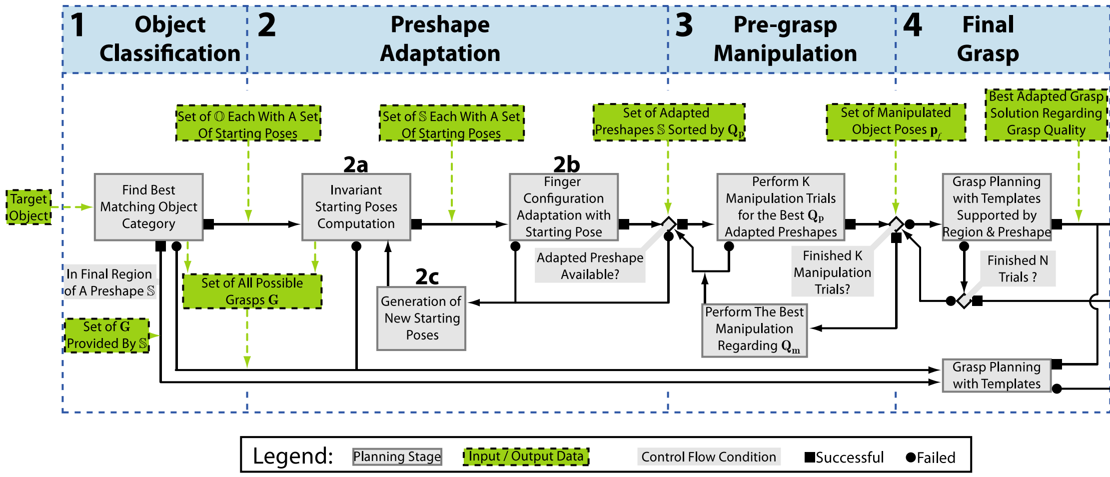

Fig. 2. The proposed pre-grasp strategy architecture with the four phases: "Object classification", "Pre-grasp Adaptation", "Pre-grasp Manipulation", "Final Grasp" embedded in the grasp planning context.

4.1.1. *Preshape*

The purpose of the *preshape* data structure is to provide a reference hand configuration in a distinct pose relative to an object. This reference preshape can be efficiently adapted to other objects within the same category to perform similar pre-grasp manipulation actions. Hence, only a small set of example preshapes is needed to be able to find suitable hand configurations and poses to perform pre-grasp manipulation actions for an object. In that context, we introduce the *preshape* data structure $\mathbb{S}$ as:

$$\mathbb{S} = (\mathbf{c}, P, C_{\mathbb{S}}, \mathbf{G}, \mathbb{M}). \tag{2}$$

Every preshape provides a hand configuration $\mathbf{c}$, which is defined by the joint values of the given robot hand. An example $\mathbf{c}$ can be determined either from human demonstration, hand crafted, or algorithmically synthesized. This work used human data to determine examples for the database. A preshape also supplies a set of starting poses P, which describe the hand position and orientation. This set P should be invariant in terms of rotation and scaling for objects within the same category. Therefore, we propose to compute P at runtime to provide appropriate starting poses $\mathbf{p}$ with respect to a given object (Fig. 2(2a)). Hence, different preshape definitions are likely for different or even the same pre-grasp manipulation actions. To be able to perform the pre-grasp manipulations $\mathbb{M}$ possible for a preshape, the hand configuration $\mathbf{c}_a$ and poses P_a adapted to the object have to satisfy constraints $C_{\mathbb{S}}$, for example finger contact with the object or mechanical joint limits. Additionally, the selection of a distinct preshape permits the selection of subset of grasp-types $\mathbf{G}$ available by the robot platform which are reasonable for final grasping. This set may include, for example, two-handed grasps or power grasps which are not feasible in the initial object pose. Finally, a given hand pose is not only a hint for adaptation, it additionally gives some insight about the manipulations that the robot can successfully execute with a high probability. Hence, we store the pre-grasp manipulations possible for each preshape as corresponding data. Although this seems redundant with respect to the data in the object category, it is crucial because multiple pre-grasp manipulations strategies might be possible for a single preshape but not necessarily for the current object category and vice versa.

4.1.2. *Pre-grasp manipulation*

The goal for a pre-grasp manipulation strategy is to relocate an object into a final region where the object is more likely to be successfully grasped. For that purpose, we propose the pre-grasp manipulation data structure $\mathbb{M}$:

$$\mathbb{M} = (a, C_{\mathbb{M}}, F). \tag{3}$$

A pre-grasp manipulation is defined by an action a, such as toppling, tumbling, rotating, pushing, or sliding. In general an action a has to satisfy constraints $C_{\mathbb{M}}$ during manipulation, for example constant object contact, force limitations of the robot joints as well as of the object surface, or object orientations such as ensuring that a cup's contents do not spill. Additionally for particular grasp-types $\mathbf{G}$ and object categories $\mathbb{O}$ there are constraints $C_{\mathbb{M}}$ for the object within the final region F such as whether the handle of a pan is reachable. The final region F is defined by the intersection of the region in which grasp-types $\mathbf{G}$, provided by the selected preshape $\mathbb{S}$, are feasible in the current robot workspace, the surface region, and the region an action is likely to succeed to relocate the object to.

To summarize, we introduced representations for our proposed data-driven pre-grasp strategy to readjust an object on a surface prior to grasping. The representations allow us to select preshapes $\mathbb{S}$ and pre-grasp manipulations $\mathbb{M}$ based on the object category $\mathbb{O}$ which is selected using the target object appearance using a classifier. These data structures build the foundation for general usage of pre-grasp strategies to increase the success rate of stable grasp acquisition.

4.1.3. *Preshape and pre-grasp manipulation for sliding*

The introduced representation for pre-grasp strategies is capable of providing solutions for different kinds of pre-grasp manipulations. Here, we demonstrate the benefit based on sliding pre-grasp manipulation and the corresponding preshapes optimized for this task.

We informally observed in the video survey mentioned in Section 3 that preshapes for sliding pre-grasp manipulation have similar initial offsets to the object surface and similar distances to object edges. Hence, a set of starting poses P can be efficiently determined for each object within a certain category based on the following tuple in the object coordinate system, illustrated in Fig. 3(a):

$$(\mathbf{x}_s, \mathbf{f}, \mathbf{d}, \mathbf{r}, \mathbf{w}) \tag{4}$$

where $\mathbf{x}_s$ describes the initial position and $\mathbf{f}$ describes the hand offset to the object surface. The original object's bounding box dimension is stored in $\mathbf{d}$ and the hand orientation stored by the roll axis $\mathbf{r}$ and yaw axis $\mathbf{w}$. The separation of the starting poses into these three parts – surface position, free-space offset, and orientation – ensure scale-invariant and rotation-invariant adaptation to objects of the same object category $\mathbb{O}$ as described in Section 4.3.

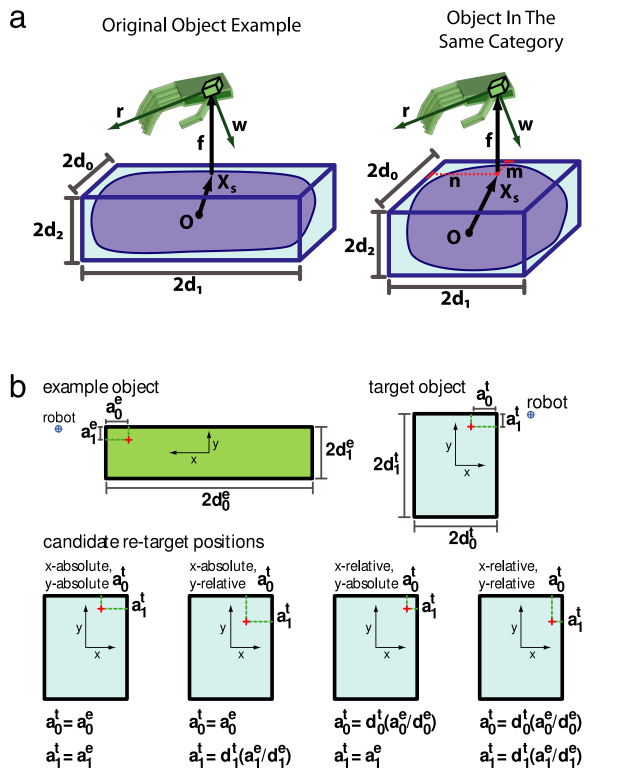

Fig. 3. (a) Representation of stored information in the preshape $\mathbb{S}$ structure. These components allow for automatic determination of the starting hand poses relative to the object. The left side shows the original object from the preshape example. The right side shows one possible starting pose made with absolute distance m and relative distance n for the two visible sides. (b) A 2D example of retargeting the starting hand pose with relative and absolute distances.

Sliding is the pre-grasp manipulation action a implemented in this paper. Hence, all preshapes have this action assigned as pre-grasp manipulation $\mathbb{M}$. Every preshape has a set of grasp-types $\mathbf{G}$ which are reasonable based on the preshape knowledge, for example a large object is preferably grasped with two hands if it is reachable for both hands. Constraints $C_{\mathbb{S}}$ for preshapes are fingertip contact with the object and collision-free arm configuration for the pose. Constraints $C_{\mathbb{M}}$ for the planning of pre-grasp manipulations are that the fingers are in contact during the whole sliding manipulation, and continuous arm configurations for the whole path can be found. Object constraints for the final object pose are not considered for $C_{\mathbb{M}}$ due to the fact that no special grasps for objects such as handled ones are available.

4.2. Architecture

Based on the presented data structures we propose a data-driven framework which performs pre-grasp strategies to grasp an object located on a surface. The general process is visualized in broad outline in Fig. 1 and in detail in Fig. 2.

At first, the method finds the best matching object category for the target object in the classification step (Fig. 2(1)) using either visual or 3D data. A more detailed description of the classifier used for our experiments can be found in Section 5.1. Detailed description of the object category $\mathbb{O}$ content can be found in Section 4.1.

The second stage is preshape adaptation, which has two major parts (Fig. 2(2a, b)). First, the pre-grasp strategy computes the set of starting poses P for the preshapes affiliated with $\mathbb{O}$. This step enables starting pose computation that is invariant to object scaling and rotation as demonstrated by our sliding preshape representation.

In the second part of the preshape adaptation stage, the kinematic template preshapes of $\mathbb{O}$ are adapted to the object surface. The *adapted preshapes* are *evaluated* to find the best adapted preshape in terms of Q_p which then is propagated to the pre-grasp manipulation phase. The rating function Q_p described in Section 5.2 checks finger contact with the object, the arm pose, the size of the final region F, and the compliance with constraints.

The next step (Fig. 2(3)) performs a *corresponding pre-grasp manipulation* $\mathbb{M}$ to the best rated $\mathbb{S}$. A successful pre-grasp manipulation plan is achieved if the object is within the final region F satisfying the constraints $C_{\mathbb{M}}$. If no successful plan is found, the best manipulation with respect to a quality metric Q_m is performed, and the search is restarted at the previous step. The metric Q_m is further described in Section 5.2, and it determines which manipulation will be performed based on the object pose, the distance to the final region, the observed constraints, and the grasp success.

In the final stage (Fig. 2(4)), the grasp-types $\mathbf{G}$ usable for a object pose $\mathbf{p}_f$ within the final region are *selected and adapted* to the object. If no grasp solution satisfying the force closure metric can be found, the previous step is repeated to manipulate the object to another location. If no grasp solution can be found within a specified number of trials, another pre-grasp manipulation or adapted preshape is selected to search for a successful final grasp.

Sections 4.3–4.6 describe the main parts of the proposed pre-grasp strategy (Fig. 2(2a, 2b, 3,4)): start pose computation, pose adaptation, pre-grasp manipulation, and final grasping.

4.3. Preshape starting pose computation

Since hand adaptation to the object surface is computationally complex and time consuming, even when limited to a selected set of preshapes or grasp solutions, it is crucial to evaluate only a limited set of starting poses. The set of starting poses must have high probability that a good adaptation based on local optimization can be found. Hence, as described in Section 4.1 every preshape has to have the ability to make a set of starting poses P available. We present in this paper an efficient way to provide P for sliding pre-grasp manipulation based on the preshape optimized for sliding introduced in Section 4.1. In the remainder of this section we refer to this distinct preshape.

To regain the starting poses P we divided the storage into three parts. Only the first part, surface position $\mathbf{x}_s$ generates a set of points (Fig. 3(b)). For every surface point an offset $\mathbf{f}$ is added and the orientation of the hand is set based on the roll axis $\mathbf{r}$ and yaw axis $\mathbf{w}$. The latter two ensure that the hand orientation is independent of the object rotation using a right-handed coordinate convention, the orientation is additionally generated correctly regardless if it is a right or left hand. To generate starting poses invariant to object pose, the object coordinate system is transformed so that the robot shoulder position has positive x, y, and z coordinates (Fig. 3(b)).

In our implementation a preshape for sliding pre-grasp manipulation is always related to three object sides. We determine the three sides with respect to the new object coordinate system. There are two ways an example position can be retargeted to a new object's side: either the absolute or the relative distance to the side can be preserved (Fig. 3(b)). The absolute and relative distance either does not or does change with scaling, respectively. The absolute distance can be measured regarding to the positive or negative side of the coordinate system, thus there are always two possibilities for absolute distances. Hence, there is one starting position if the relation to all three sides is relative. If the relation to one side is measured absolute, there are 6 different solutions one shown in (Fig. 3(a)), for every side two distances. For two absolute distances there exist 12 possible solutions, and if all sides distances are retargeted as absolute 8 possible starting positions are available. In total 27 starting positions $\mathbf{x}_s$ are necessary to express all possible relations of the surface point with the object sides. As an example we show 4 possible starting positions for a 2D example in Fig. 3(b).

Thus, multiple surface positions are available, and this overhead is acceptable in order to store a set which contains a promising relative pose for a new object. Additionally, unreachable solutions or starting poses initially in collision with the object are automatically discarded in the next step of planning.

4.4. Finger configuration adaptation and evaluation

The second phase of preshape adaptation (Fig. 2(2b), detailed in Fig. 4) builds upon the set of preshapes $\mathbb{S}$ selected by the object category and P determined by the previous step. The key aspect is that preshapes as proposed in Section 4.1 enable efficient computation of hand configurations suitable for pre-grasp manipulations considering environmental, robot, and object constraints.

Hence, the goal of this phase is to find suitable hand poses and configurations for the current object to perform related pre-grasp manipulations. Several solutions for this subtask are available in literature. For example [44] suggested a grasp adaption algorithm based on mapping grasp positions from an example to a new object, using an object similarity measure. However, for the usage in this framework to pre-grasp manipulate the object, the synthesized grasps do not have to be as similar to the example object as possible. They have to respect the original association to global object features such as the object sides, which is why we propose this new adaptation method. Hsiao and Lozano-Perez [19] proposed a solution for template grasp adaptation for direct grasping using the start position mapped onto related object. In contrast to Hsiao and Lozano-Perez [19] we generate multiple starting poses to be rotation and scaling invariant, and for each of these starting poses we perform a local optimization to adapt the hand to the object.

Since a grasp position is needed to perform a pre-grasp manipulation strategy and not to direct grasp the target object, we present a new adaptation algorithm which locally optimizes the hand configuration using additional context knowledge provided by the preshape structure. The data representation provides starting poses and corresponding hand configurations with a high probability for successful adaptation. Because of the relatively expensive adaptation step we first check each starting pose for current reachability using an inverse kinematics (IK) checker for the arm. Hence, we do not initially search for starting preshapes, but we evaluate the preshapes which are reachable. To evaluate them, we need to adapt the preshapes to the current object.

To adapt the hand configuration to the given target object the hand position is only changed in the initial step (Fig. 4(2b)), such that the hand is in collision with the target object. Due to the configuration $\mathbf{c}$ given by preshape $\mathbb{S}$, the hand is already in a promising configuration for contacts. Therefore, we individually search for finger contact with the object surface based on an iterative inverse kinematics approach for the current finger chain. This approach prevents awkward hand configurations because even if no fingertip contact is made the fingertips are close to the object surface. Due to the local optimization the initial finger configuration ensures natural looking configurations. This method is described in detail in Fig. 4(2b). Once the preshapes have been adapted to the object, we then score them by a rating function Q_p regarding the pre-grasp manipulations they correspond to. An ordered set of adapted preshapes, sorted by rating, is provided to the next stage if the score of the preshapes is higher than a threshold. If none of the selected preshapes can be adapted or none fulfill the quality metric Q_p with a score higher than the threshold, new starting poses are generated. In these cases (Fig. 4(2c)), we use the original preshape configuration $\mathbf{c}$, and then we locally optimize the hand pose $\mathbf{p}$ based on the closest surface points to the active fingertips. After the hand is iteratively moved to the new pose, the process of finding fingertip contact restarts (Fig. 4(c)).

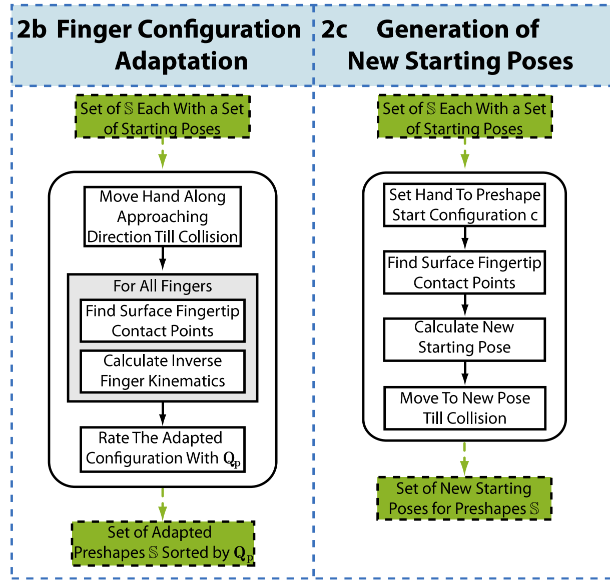

Fig. 4. The left side of our proposed algorithm adapts a hand configuration of a given preshape to the object surface. If no adapted preshape with a high enough rating Q_p is available, the right side of the algorithm computes new starting poses.

4.5. Pre-grasp manipulation

Currently, the pre-grasp strategy has achieved the following goals: A set of preshapes has been adapted to the object, and the best adapted preshape relative to a quality function Q_p has been selected (Fig. 2(2)).

Now a possible pre-grasp manipulation strategy has to be selected (Fig. 2(3)). In general a preshape may allow multiple pre-grasp manipulations and additionally is used in multiple object categories. Hence, to select the possible pre-grasp manipulations of an adapted preshape and the object category, only those that are possible for both are allowed, as stated in Eq. (5). This is because it might be that a distinct object category has more restrictions to manipulations than the selected preshape or vice versa.

$$\mathbb{M}_f = \mathbb{O}.\mathbb{M} \cap \mathbb{S}.\mathbb{M}. \tag{5}$$

Computation of the preshape manipulation trajectory is carried out using a standard planner (see Section 5). With completion of this phase, the object is located within the final region and satisfies the pose constraints, and thus the grasping action can be planned more easily.

4.6. Final grasp

The final grasp is planned once the pre-grasp manipulation has successfully located the object within a final region F respecting constraints $C_{\mathbb{M}}$. Finding grasp candidates for an object located in such a region has been well explored in robotics and computer graphics. Literature such as [15,18,19,32,33] about this topic was introduced in Section 2. If a successful grasp solution respecting force closure metric is found, the pre-grasp strategy is finished.

Due to the data-driven approach proposed in this paper, additional knowledge is available for planning the final grasp. In a template approach, a small set of template grasps can be automatically selected based on the previously-selected preshape. The final region to where the target object is relocated has a corresponding statistical model describing the grasp templates that are possible within this area, as expressed by Eq. (6).

$$\mathbf{G}_f = \mathbb{S}.\mathbf{G} \cap (g \in F). \tag{6}$$

More details about the actual grasp planner implementation used for the experiments presented in this paper can be found in Section 5.4.

If no stable grasp solution is found (Fig. 2(4)), there are four possible options our framework proceeds to test. First, try the same manipulation toward another pose within the final region. Second, change the manipulation strategy possible for the preshape and object category. Third, change the possible preshape to a worse-rated one. Finally, admit that no possible solution for stable grasping can be found.

5. Implementation

We implemented our framework as a plugin in OpenRAVE [45] which provides collision checking, inverse kinematics (IK) solutions and Rapidly-exploring Random Tree (RRT) planning to our simulation. The RRT motion planner is used to find a continuous motion between two distinct hand configurations in a certain pose.

For the sliding preshapes and pre-grasp manipulation the following assumptions are made. First, surface geometry models are available for all objects. Object weight is known a priori, and objects have an initial coordinate system with a known upright axis indicating how it rests on the surface. We assume that as long as fingertip contact is made during manipulation, the object is moving with the hand in reality as in simulation. For real execution this can be ensured through force control which was performed on the humanoid robot platform described in Section 6.5. Additionally, our sliding implementation does not support obstacles on the surface. Other pre-grasp manipulation strategies as well as a more sophisticated sliding implementation are envisioned.

5.1. Object classification

In our implementation we use the dimensions, weight and curvature as an input vector for a multilayer neural network. The dimensions and curvature are computed online based on the object triangular mesh, and the object weight is assumed as a priori knowledge. This classification results in assigning an object category stored in a database to the given target object. The classifier was trained offline with two manually-selected object examples for every object category.

The goal of the object classification is to select the best matching object category for a given object so that the proposed pre-grasp strategy can prepare the object pose for final grasping. Also note that our method will not necessarily fail in response to misclassification of objects, due to the adaptation process that is described in Section 4.4. The purpose of the data provided by the object category was described in detail in Section 4.1, whereas a distinct manifestation used for evaluation is presented in Section 6.1.

5.2. Rating function

As introduced in Section 4.4 the adapted preshapes are evaluated by the corresponding Q_p. We propose a rating function which prefers more finger contacts, a large final region size, as many pre-grasp manipulations as possible, and unconstrained arm solutions regarding joint limits:

$$Q_p(\mathbf{c}, \mathbf{p}, \mathbf{q}, F, C_{\mathbb{S}}) = \alpha_1 \sum_{i=1}^{n} b_i + \alpha_2 size(F) + \alpha_3 sat(C_{\mathbb{S}}) + \alpha_4 \sum_{j=1}^{a} \left(\frac{(\min_j - q_j)^2 + (\max_j - q_j)^2}{(\min_j - \max_j)^2} \right)^{-1} \tag{7}$$

where n is the number of fingers, b_i is a binary indicator expressing whether the finger i made contact, and a the number of arm joints. The relative size of the final region F is determined by $size(F)$. The constraints $C_{\mathbb{S}}$ are checked by a binary function $sat(\cdot)$ to determine if a related pre-grasp manipulation can be performed. For example, one constraint is whether the force on the object is sufficiently below the maximum contact forces of the fingers. The values $\min_j$ and $\max_j$ denote the joint limits, and q_j is the current joint value. The different parts of the equation are weighted by α_k.

A second rating function Q_m, first introduced in Section 4.2, is used to refine the starting pose when there is no pre-grasp manipulation found that satisfies the constraints. The function Q_m evaluates starting poses for a new adaptation set:

$$Q_m(\mathbf{p}_f, F, C_{\mathbb{M}}, g) = \beta_1 d(\mathbf{p}_f, F) + \beta_2 d(\mathbf{p}_f, C_{\mathbb{M}}) + \beta_3 g \tag{8}$$

where $d(\mathbf{p}_f, F)$ is the distance to the final region F and $d(\mathbf{p}_f, C_{\mathbb{M}})$ is the distance to the object pose constraints.

It is possible that the object could be within the final region but not graspable by any grasp type $g \in \mathbf{G}_f$ representing e.g., two-handed side grasps or one handed power grasps. In this case, a correction can be made by removing the final region for a particular grasp-type or reducing the likelihood for choosing this final region for later trials. Our implementation does not include these correction approaches to address these cases, which were rare in our experiments.

5.3. Pre-grasp manipulation

Based on the observation within the video survey and human motion capture experiments mentioned in Section 3, we implemented sliding manipulation as the pre-grasp manipulation strategy of choice. Alternative manipulation strategies include toppling previously investigated by Lynch et al. [29] and pre-grasp rotation investigated by Chang et al. [22,24,42]. The approach of push-grasping proposed by Dogar et al. [46] tries to push/slide the object within the hand to be able to grasp the object. In contrast to the push-grasping proposed by Dogar et al. that makes local adjustment to object position and is limited to the approach direction of the push-grasp primitive, our proposed planner can relocate the object over greater distances and directions on the surface such that the following grasp planning can be performed more efficiently.

To relocate the object to better grasping configurations, our method must generate reachable random locations within the final region. These regions are provided by the selected pre-grasp sliding manipulation. To find reachable locations, the hand orientation has to be altered within the final region. If a kinematically-reachable and collision-free end pose solution is found, then the planner attempts to generate an arm motion trajectory from the start to the synthesized end pose. This is done through computing inverse kinematic solutions for interpolated hand poses including the orientation. Candidate trajectories are checked for necessary constraints, e.g. finger contact during the complete trajectory. To ensure continuity of the trajectory the joint values between two discrete steps of the trajectory are allowed to change within a certain threshold. As stated before, the simulation assumes that the friction between the hand/object is higher than that between the object/table, which was a reasonable assumption for the experiments described in Section 6.5. If a feasible trajectory is found, the final grasp planning stage described in Section 5.4 is started. Otherwise a new random location is generated or another starting preshape is selected until there is a feasible trajectory.

5.4. Final grasp

The benefit in our framework is that object knowledge is already available through selected template grasps and, more importantly, the object is in a better location for grasping. Berenson et al. [32] proposed a pre-computation of possible grasp candidates sampling the object surface to gather starting positions to adapt template grasp. Due to the better location and provided template grasps based on object knowledge given by the selected preshape $\mathbb{S}$ Berenson's offline planner can be enhanced using an iterative sampling process to generate starting poses. In each iteration the template grasps are adapted, as proposed in Section 4.4, to the object beginning at these starting poses. If no stable grasp solution can be found within one iteration the sampling precision is refined and the adaption process starts again. This method of course does not find the best grasp but because of the superior object pose a small set of robust template grasps can be adapted and as soon as a stable grasp solution is found it can be integrated in the plan and executed on the real robot. A stable solution in this context has to satisfy the force-closure metric proposed by Ferrari et al. [35]. Another grasp planner used for evaluation is proposed by Przbylski et al. [33] using the medial axis to generate grasp hypothesis. This planner was used for the experiments on the humanoid robot platform ARMAR-IIIb described in Section 6.5. Hence, we showed the customizability of the framework with different state-of-the-art grasp planners which furthermore benefit from the context knowledge and better object pose.

6. Experiments and results

We compared our method in simulation to a traditional planner for direct object grasping. We evaluated the success rate of finding a feasible object acquisition plan and the computation time.

6.1. Scenario

In the test scenario, there is a single table in front of the robot. The robot is a bi-manual humanoid model from OpenRAVE [45] with 7-DoF arms and 15-DoF hands (Fig. 1). In our implementation, the right hand has been replaced with the Shadow Hand (Shadow Robot Company, London, UK) with 23 DoFs.

Two objects from each of the four sliding manipulation preshape sets were tested (Table 1). Each preshape set consists of five preshapes from two objects in the same category, which were manually extracted from human examples gained with a motion capture experiment. The tested objects were not included in the database examples. For the final grasps we provided 12 grasp-types to both planners. Our method selects suitable grasp-types corresponding to the selected and adapted preshape. The direct grasp planner randomly selects one out of the 12 grasp-types and tries to find a stable lifting grasp, until a solution is found or all 12 are tried. We limited the search for a stable solution for each grasp-type to 20 s.

Each object is placed at a random position and plane orientation on the table within a 1.2 m × 0.6 m region in front of the robot, shifted 0.4 m right of the robot center. We discarded the random position if the object is not within the reaching radius of the robot's right arm. We generated poses in this manner to obtain 36 reachable poses per object. Both planners attempted an object acquisition solution for the same 36 starting object poses.

Table 1
Objects and object categories[a].

$\mathbb{O}_0$ flat shape	$\mathbb{O}_1$ light box	$\mathbb{O}_2$ heavy box	$\mathbb{O}_3$ cylinder
CD	Baseball bat	Dictionary	Cookie tin
Credit card	Stapler	Keyboard box	Sugar canister
15-cm ruler	Tape dispenser	VGA-splitter	Plant pot
House key	Cassette tape	Food storage	Water jug
	Bed linen box	Container/box	
	Hard-drive		

[a] All objects were manipulated in the human study. Objects in the first row, and other objects, were tested in the simulation validation. The objects in the second and third rows provided the examples for preshape and manipulation information in the database.

Table 2
Simulation results for the method comparison.

	Successes out of 36	Mean planning times (s)		
		Pre-grasp	Grasp	Trajectory
$\mathbb{O}_0$: CD, ruler				
Pre-grasp push	36	1.5	2.0	15.0
Direct grasp	1	–	53.4	0.4
$\mathbb{O}_1$: bat, stapler				
Pre-grasp push	34	3.4	15.2	16.7
Direct grasp	25	–	15.8	11.8
$\mathbb{O}_2$: book, food				
Pre-grasp push	36	1.5	3.2	17.7
Direct grasp	26	–	16.5	8.4
$\mathbb{O}_3$: tin, jug				
Pre-grasp push	36	2.1	2.6	18.6
Direct grasp	25	–	33.6	8.9
Total	Out of 144	Pre-grasp	Grasp	Trajectory
Pre-grasp push	142	2.1	5.6	17.0
Direct grasp	77	–	22.3	9.6

6.2. Direct grasp planner

For direct grasp planning the implementation described in Section 4.6 was used but because no object context was available, grasp-types available for the robot were randomly selected and adapted.

6.3. Simulation results

Table 2 presents the results, separated for each object category and planning phase. The "Pre-grasp" column for our approach includes the object classification, calculation of the preshape starting poses, preshape adaptation and evaluation, and pre-grasp manipulation. The "Grasp" column contains the final grasp adaptation for both our method and the direct grasp planner. The "Trajectory" column consists of the trajectory plan from the initial position to the adapted final grasp for both methods: direct grasp planning, as well as for our approach if the object is initially in the final region F of the preshape. If not, in our approach the trajectory consists of two separate trajectories, one from the initial pose to the adapted sliding preshape and a second one from final sliding to the final grasp pose.

Our strategy increased the success rate for object acquisition for all tests as shown in Table 2 and Fig. 5. In addition, our approach reduced the computation time for grasp adaptation significantly regardless of whether the object is directly graspable or not (Fig. 6).

Fig. 7 shows example simulation results for different object categories.

6.4. Perceptual evaluation

We also evaluated human response to the pre-grasp manipulation plans and direct grasping plans. In our survey, 21 participants viewed pairs of simulation videos showing the humanoid agent using, in a random order, either pre-grasp manipulation or direct

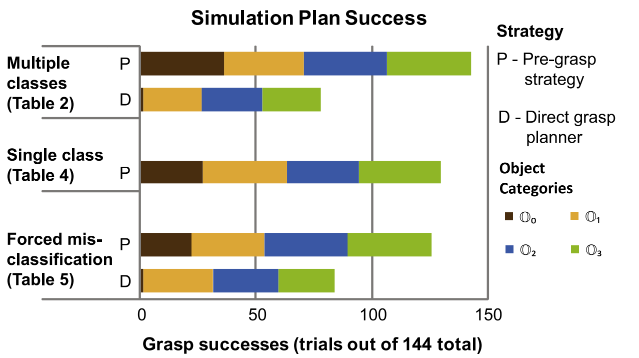

Fig. 5. Summary plot of planning success comparison for simulated manipulation plans. See Tables 2, 4 and 5 for numerical results.

grasping. Participants selected the preferred video in each pair. Table 3 shows that pre-grasp manipulation was preferred by more people for the cookie tin, baseball bat, and linen box objects. A chi-square test on the number of participants preferring pre-grasp manipulation or direct grasping for at least 3 of the 5 video pairs rejected that the ratio was balanced 50–50 ($p(X^2 = 5.76, df = 1) = 0.02$).

6.5. Physical demonstration

We demonstrated the physical plausibility of our simulated pre-grasp strategy plans on two multi-fingered robot manipulators (Fig. 8). The first system consists of a 9-DoF kinematic chain for setting the hand pose for the attached Shadow Hand robot with 5 fingers, factorized into the 7-DoF Motoman arm and 2-DoF Shadowhand robot wrist. The second setup uses the ARMAR-IIIb [1] humanoid robot with 8-DoF kinematic chain to control the hand pose, decomposed into a 1-DoF hip and a 7-DoF arm. The anthropomorphic hand of ARMAR-IIIb has 8-DoF.

In our first example demonstration on Motoman, the object is a CD, which is difficult to grasp from a table because of its thin edge (Fig. 8(a)). However, the Motoman with Shadow Hand was able to grasp the CD after first using a sliding pre-grasp manipulation planned with our method. The CD was manually placed on the table to match the simulated task scene. The Motoman arm trajectory produced by our simulation method was executed open-loop on the robot. Due to limitations of the control synchronization, the hand preshapes for the Shadow Hand were selected from our simulated plan but were manually pre-set to match the arm trajectory timing.

In our second example demonstration on ARMAR-IIIb, a pizza box was placed on a table which is hard to grasp if the box rests completely on a table (Fig. 8(b)). Therefore ARMAR-IIIb detected the target object and sent the object pose to our framework which then generated a plan executed on the robot. The contact during sliding pre-grasp manipulation was ensured using force control with force torque sensors in the wrist. After the pre-grasp manipulation was executed ARMAR-IIIb searched for the object again and updated the current pose in simulation which then

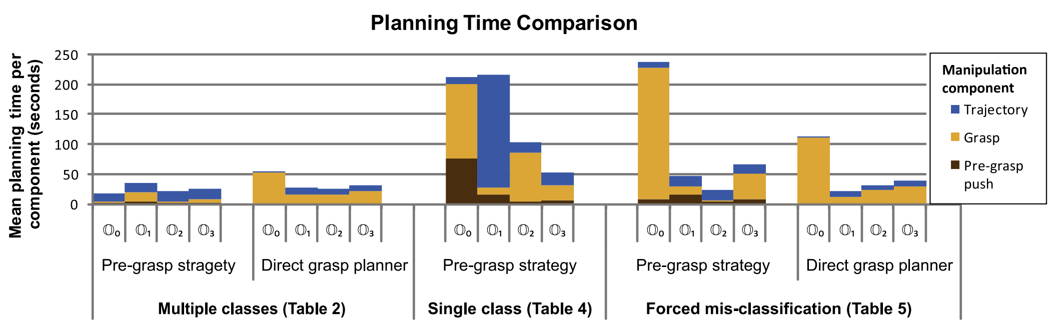

Fig. 6. Summary plot of planning time comparison for simulated manipulation plans. See Tables 2, 4 and 5 for numerical results.

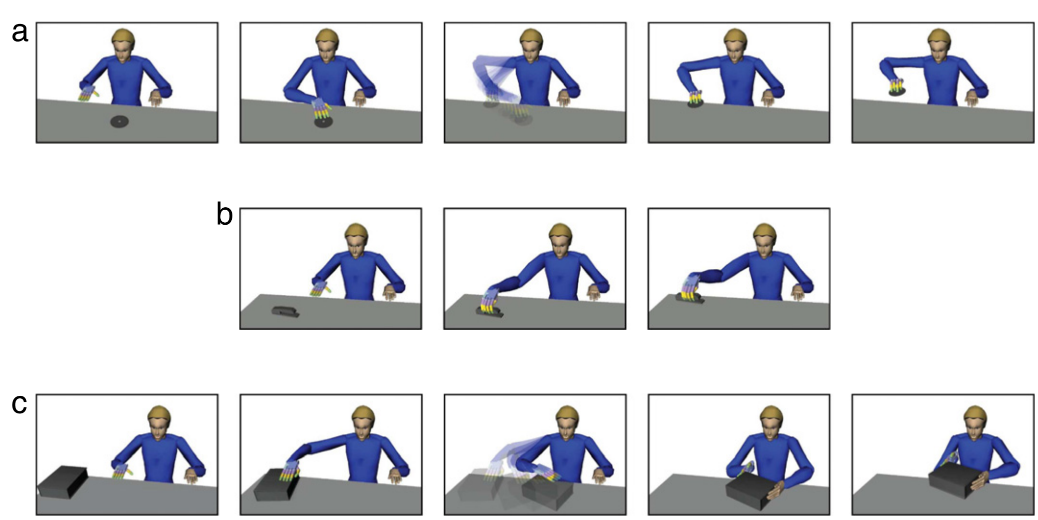

Fig. 7. Example simulation results synthesized from our template-based system. (a) Acquisition of a CD using pre-grasp sliding. Direct grasps of the CD were difficult to find because of the thin object profile. Pre-grasp sliding adjusted the CD location to the edge of the table so that the bottom surface is graspable. (b) Acquisition of a stapler. The initial location of the stapler was within the final grasping region associated with the best matching object template. Thus a direct grasp was attempted and succeeded, and no pre-grasp interaction was synthesized for this trial. (c) Acquisition of a thick dictionary book. Direct grasps were difficult to reach because the dimensions of the book cover were large and the initial position was far to one side of the robot. Pre-grasp sliding to the final region in front of the robot enabled a final bi-manual grasp of the heavy book.

Table 3
The number of participants who preferred either pre-grasp interaction or direct grasping in the video survey.

Object	Manipulation method	
	Direct grasp	Pre-grasp
Cookie tin	3	18
Dictionary	10	11
Baseball bat	6	15
CD	9	12
Linen box	5	16
Overall preference ($\geq$3/5 objects)	5	16

determined whether to either compute the final grasp or perform another pre-grasp manipulation.

The video at http://his.anthropomatik.kit.edu/english/532.php shows the comparison between the simulated Motoman/ARMAR-IIIb plan and the physical execution on the robots.

7. Validation of template classification

The experiments in the previous section demonstrated the utility of augmenting the grasping process with pre-grasp interaction for object acquisition. Using our framework for planning pre-grasp sliding interactions, we were able to synthesize plausible and natural-looking actions that improved the reachability of the object for grasping.

We now examine how both the initial template classification and the later template adaptation are critical for finding pre-grasp interaction plans within a tractable time. To validate these stages, we consider experiments with two modified versions of the method presented in Section 4.

First, we test the value of organizing examples according to the object context. We modify our framework by eliminating object classification while retaining the information from the same examples of pre-grasp hand shapes and poses relative to the objects. In essence, this reduces the classification step to predicting a single class that includes all of the example data. Thus we still use the examples to find promising hand pre-shapes, but the selection is agnostic to the object features.

Second, we test the robustness to misclassification of the object class. In this set of experiments, the examples are organized according to the same four object classes described in Table 1. However, we deliberately mis-assign the classification result in order to investigate the response of the hand adaptation process.

7.1. Example organization in a single-class

In this experiment, we keep our original framework but instead input a database consisting of a single class which includes all example hand preshapes. This is equivalent to omitting the classification of the object features into an object category to determine a subset of the examples. Thus the candidates for initial hand poses and preshapes are selected by evaluating and attempting adaptation with all examples in the database in a random order. The final grasp shape and final region for object location are selected in the same manner as before, that is according to the corresponding preshape that is attempted during the pre-grasp pushing manipulation.

Due to the increase in the number of examples to test, an additional termination criterion is included to constrain the length of the experiment. The planning process was terminated after a total time of 15 min when no successful grasp for object acquisition was found. This prevents the experiment from testing every example to exhaustion. Only the pre-grasp interaction strategy was tested, without the direct grasping comparison.

Table 4
Simulation results using a single class template. The average results from the original multi-class template experiments reported in Table 2 are included again for comparison.

	Successes out of 36	Mean planning times (s)		
		Pre-grasp	Grasp	Trajectory
$\mathbb{O}_0$: CD, ruler				
Pre-grasp push	27	75.7	124.0	11.6
$\mathbb{O}_1$: bat, stapler				
Pre-grasp push	36	16.0	10.6	189.7
$\mathbb{O}_2$: book, food				
Pre-grasp push	31	3.2	83.0	17.8
$\mathbb{O}_3$: tin, jug				
Pre-grasp push	35	5.8	25.3	22.4
Total	Out of 144	Pre-grasp	Grasp	Trajectory
Pre-grasp push	129	22.7	55.7	65.7
Total repeated from Table 2, with multiple template classes:				
	Out of 144	Pre-grasp	Grasp	Trajectory
Pre-grasp push	142	2.1	5.6	17.0
Direct grasp	77	–	22.3	9.6

The results in Table 4 indicate that the lack of organizing examples by object class decreased the success of finding feasible grasping plans by 9% compared to the original framework results in Table 2. In addition, the average computational time required for planning the manipulation action increased by 11-fold, 10-fold, and 4-fold for the pre-grasp interaction pushing, the final grasp planning, and the arm trajectory planning. These results show the benefit of having examples organized in a manner that allows the identification of promising candidate configurations.

7.2. Results for object category misclassification

In this second validation experiment, we test the adaptation phase of our framework.

The previous validation experiment demonstrated the need for example organization into template categories. In our original experiments presented in Section 6, the classification results appeared reasonable even for objects that did not necessarily fit the semantic labels we used in Table 1. For example, the water jug object has a square cross section in the bottom half of its the base, but it was classified in the "cylinder" category $\mathbb{O}_3$ not the "heavy box" category $\mathbb{O}_2$. This resulted in natural-looking pre-grasp interaction where the hand contacted the side surfaces of the jug similar to the lateral surfaces of a cylinder, instead of interaction with hand contact at the edge between the top and side faces of a box.

For additional novel objects or a different object classifier, however, it is possible that the classification method may result in a low confidence between two or more classes that would be similarly appropriate for the object. Here we test how the adaptation phase in our framework can be used to robustly modify the hand preshapes to new objects when the classification results are different.

In this experiment, our framework remains unchanged except the output of the classification result. We force the classification result to swap between two pairs of categories:

- if the original classification result would have been $\mathbb{O}_0$ for thin objects, the mis-classified result is output as $\mathbb{O}_1$ for light weight boxes, and vice versa
- if the original classification result would have been $\mathbb{O}_2$ for heavy boxes, the mis-classified result is output as $\mathbb{O}_3$ for cylinders, and vice versa.

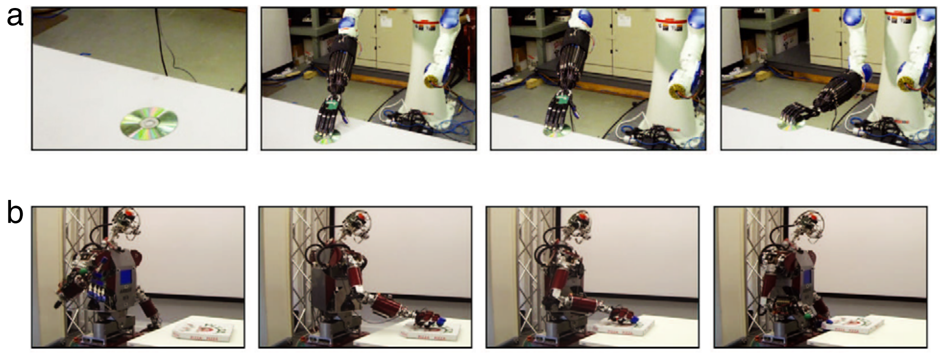

Fig. 8. (a) The Shadow Hand manipulator uses pre-grasp interaction to slide a thin CD to the table edge before grasping. (b) The ARMAR humanoid uses pre-grasp pushing to reach a pizza box with a one-handed grasp of the box height instead of a two-handed grasp of the box width.

We did not consider the extreme misclassification results of swapping, e.g., $\mathbb{O}_0$ for thin objects with $\mathbb{O}_2$ for heavy boxes, since this result is unlikely if the classifier has been trained on sufficient examples.

The results in Table 5 for the misclassification tests indicate that there was a decrease in the success rate for finding feasible grasp plans, compared to the "correct" classification results in Table 2 (125 successful plans overall instead of 142). Interestingly, the direct grasp successes increased for the misclassification results. This was due to the fact that the final region for the object grasping is associated with the mis-classified object class. Thus, sometimes a partial pushing interaction is initiated that moves the object from the starting location. For direct grasping, there is no object motion from the start location, and the misclassification in some examples resulted in different hand preshapes being used for successful grasping.

The comparison for the timing results (Table 5) with the original computation time averages (Table 2) indicates that the main increase in planning time occurred in the grasp planning phase, along with an increase in the pre-grasp push planning time. This is due to the association of the final grasp hand shapes with a selected pushing preshape. This association leads to time-efficient selection and natural-looking manipulation actions when the object is "correctly" classified. However, in the misclassification case, the associated final grasp shapes would not be appropriate for the object. This point in fact demonstrates the utility of using pre-grasp interactions because the hand shapes for pushing or other pre-grasp interactions are usually less restricted than the final grasps required for lifting an object. That is, the adaptation of a pushing shape primarily needs to make contact with the object to exert forces in the right direction, but the grasping shape must satisfy more constraints to be a feasible lifting grasp. In systems where the classification confidence is low, pre-grasp template adaptation using the selected class may still be reasonable, but the final grasping phase could be altered to consider hand shapes from similar grasp classes instead of only the selected class.

Table 5
Simulation results for mis-classified template examples. Note the changes in object class label $\mathbb{O}_i$ compared to Table 2. The average results reported in Table 2 for the original experiments with correct classifications are included again for comparison.

	Successes out of 36	Mean planning times (s)		
		Pre-grasp	Grasp	Trajectory
$\mathbb{O}_1$: CD, ruler				
Pre-grasp push	22	7.4	220.8	8.8
Direct grasp	01	–	111.3	0.4
$\mathbb{O}_0$: bat, stapler				
Pre-grasp push	31	16.6	12.2	18.2
Direct grasp	30	–	12.5	8.8
$\mathbb{O}_3$: book, food				
Pre-grasp push	36	3.3	2.8	17.2
Direct grasp	28	–	22.7	8.8
$\mathbb{O}_2$: tin, jug				
Pre-grasp push	36	2.1	2.6	19.7
Direct grasp	24	–	30.0	8.9
Total	Out of 144	Pre-grasp	Grasp	Trajectory
Pre-grasp push	125	7.0	43.4	16.7
Direct grasp	83	–	22.2	8.7
Total repeated from Table 2, with correct template classification:				
	Out of 144	Pre-grasp	Grasp	Trajectory
Pre-grasp push	142	2.1	5.6	17.0
Direct grasp	77	–	22.3	9.6

8. Discussion

In this paper we presented a framework for representing and re-synthesizing examples of pre-grasp interaction, particularly sliding actions, that can improve grasping success. Our approach was based on patterns observed in human demonstration of pre-grasp interaction, where the choice of hand position relative to the object and hand shape were similar for similar object features. These examples simplify the high-dimensional search for candidate hand configurations by providing promising templates for a new object based on its object category. With this reduction of candidate configurations, the search becomes tractable for articulated manipulators with multi-fingered hands.

The basis of our framework is the representation of a pre-grasp interaction that includes context information including the hand preshape and pose as well as other constraints such as the final region suitable for the final grasp. Another key component of our method is the representation of the candidate preshape poses relative to the object, which are stored and regenerated to account for changes in object scaling and rotation. Altogether, these pre-grasp manipulation actions, which include all kinds of prior adjustments to object acquisition, and final grasping can be computed online. The whole strategy results in more robust and stable object grasping.

8.1. Organization of examples

Our validation of the proposed framework focused on the importance of organizing examples for efficient re-use. In particular, when there are several examples of candidate hand preshapes, it is not sufficient that the examples exist in the reference database. Instead, organization – in this case by object categories – allowed our method to quickly determine a subset of examples that were suitable for the simulation tasks. The efficiency of identifying promising templates is an aspect of example-based planning that will be even more critical for larger databases of manipulation actions.

In our current implementation, the database of preshape examples was manually organized based on our observation from the human motion capture studies. We found that examples from a small set of 8 objects and 4 participants provided promising templates that could be adapted successfully to new objects. An extension that is beyond the scope of the current work is to extend the system to include automatic learning or updates of the object classes and examples. This can be done with both new human-demonstrated examples as well as examples that come directly from the robot's manipulation that result from our synthesis method.

8.2. Directions for future extensions

In future work, we plan to extend our database and representations to accommodate additional types of pre-grasp manipulation such as tumbling or toppling maneuvers. Our framework was designed to be able to support other action modes in addition to sliding and pushing interactions within the template preshape data structures. Another aspect for future development is a more sophisticated version of the sliding interaction that includes obstacle avoidance of additional objects on a cluttered table surface.

Future steps may explore reducing the assumptions about the friction coefficients in the environment. Initially planning with friction assumptions, but then using feedback during the pre-grasp manipulation to obtain new object parameters such as inertia and friction coefficients would increase the stability of the final grasp due to the extra knowledge gained. The material parameters could be estimated or refined in a preliminary interaction with the object or by learning relationships from visual characteristics such as reflectance or texture to identify the object material.

Interesting enhancements to completely realize the proposed planning framework include fully automatic extraction of preshapes based on human motion data, as well as automatic generation of final regions in simulation and real robot experiments based on statistical learning methods. In addition, if enough object categories and pre-grasp manipulations are available in the database, the object categories could be learned based on the possible pre-grasp strategies and grasps in combination with the object appearance.

Another interesting idea is to parallelize different planning steps. As soon as the object representation is selected, all possible preshapes are available. Hence, one approach is to evaluate final grasp poses in parallel which is done in [32] by Berenson et al. as offline precomputation. Preshape adaptation can also be parallelized due to orthogonal usage. This would speed up final grasp planning because it is only a selection of possible grasp solutions with respect to environmental restrictions.

Extending the database with more object examples would give more insight for an useful object category distribution. Then the existing categories either have to be refined and new ones created, or more preshapes have to be generated for the broad categories. Having more object categories would result in more complex classification which additionally would need more information. On the other side, more preshapes within an object category will increase planning time. These two effects should be balanced to find the ideal ratio between the object category size and the number of preshapes needed to robustly perform pre-grasp manipulation.

Overall, the proposed representation of pre-grasp strategies for object manipulation significantly increases the object acquisition success rate. This underlines the great potential of using human behavior knowledge to develop new planning strategies.

Acknowledgments

The first author received support from the InterACT Exchange project at Carnegie Mellon University.

The second author's work was supported by the National Science Foundation under Grant #1019343 to the Computing Research Association for the CIFellows Project. The author also received support as a NASA Harriett G. Jenkins Pre-Doctoral Fellow while at Carnegie Mellon University.

The third author's work was supported by the National Science Foundation (CCF-0702443).

The fourth and fifth authors' work described in this paper were partially conducted within the EU Cognitive Systems project GRASP (IST-FP7-IP-215821) funded by the European Commission and the German Humanoid Research project SFB588 funded by the German Research Foundation (DFG: Deutsche Forschungsgemeinschaft).

Appendix. Supplementary data

Supplementary material related to this article can be found online at doi:10.1016/j.robot.2011.07.015.

References

[1] T. Asfour, K. Regenstein, P. Azad, J. Schroder, A. Bierbaum, N. Vahrenkamp, R. Dillmann, ARMAR-III: an integrated humanoid platform for sensory-motor control, in: Proc. IEEE Int. Conf. Humanoid Robots, Humanoids, 2006, pp. 169–175.
[2] T. Wimbock, C. Ott, G. Hirzinger, Impedance behaviors for two-handed manipulation: design and experiments, in: Proc. IEEE Int. Conf. Robotics and Automation, ICRA, 2007, pp. 4182–4189.
[3] K. Okada, T. Ogura, A. Haneda, J. Fujimoto, F. Gravot, M. Inaba, Humanoid motion generation system on HRP2-JSK for daily life environment, IEEE International Conference Mechatronics and Automation, 2005, pp. 1772–1777.
[4] S. Haykin, Neural Networks: A Comprehensive Foundation, Prentice Hall, 2008.
[5] C. Burges, A tutorial on support vector machines for pattern recognition, in: Data Mining and Knowledge Discovery, Springer, 1998, pp. 121–167.
[6] M.R. Cutkosky, On grasp choice, grasp models, and the design of hands for manufacturing tasks, IEEE Journal of Robotics and Automatation (1989) 269–279.
[7] J.R. Napier, Hands, Princeton University Press, Princeton, New Jersey, 1993.
[8] S.J. Edwards, D.J. Buckland, J.D. McCoy-Powlen, Developmental & Functional Hand Grasps, Slack Incorporated, Thorofare, New Jersey, 2002.
[9] K. Ikeuchi, T. Suehiro, Toward an assembly plan from observation part I. Task recognition with polyhedral objects, IEEE Journal of Robotics and Automatation (1994) 368–385.
[10] S.B. Kang, K. Ikeuchi, Toward automatic robot instruction from perception-mapping human grasps to manipulator grasps, IEEE Journal of Robotics and Automatation (1997) 81–95.
[11] S. Ekvall, D. Kragic, Grasp recognition for programming by demonstration, in: Proc. IEEE Int. Conf. Robotics and Automation, ICRA, 2005, pp. 748–753.
[12] K. Bernardin, K. Ogawara, K. Ikeuchi, R. Dillmann, A sensor fusion approach for recognizing continuous human grasping sequences using hidden Markov models, IEEE Transactions on Robotics (2005) 47–57.
[13] H. Kjellstrom, J. Romero, D. Kragic, Visual recognition of grasps for human-to-robot mapping, in: Proc. IEEE Conf. Intelligent Robots and Systems, IROS, 2008, pp. 3192–3199.
[14] S. El-Khoury, A. Sahbani, V. Perdereau, Learning the natural grasping component of an unknown object, in: Proc. IEEE Conf. Intelligent Robots and Systems, IROS, 2007, pp. 2957–2962.
[15] A. Miller, S. Knoop, H. Christensen, P. Allen, Automatic grasp planning using shape primitives, in: Proc. IEEE Int. Conf. Robotics and Automation, ICRA, 2003, pp. 1824–1829.
[16] C. Goldfeder, P. Allen, C. Lackner, R. Pelossof, Grasp planning via decomposition trees, in: Proc. IEEE Int. Conf. Robotics and Automation, ICRA, 2007, pp. 10–14.
[17] Y. Li, N. Pollard, A shape matching algorithm for synthesizing humanlike enveloping grasps, in: Proc. IEEE Int. Conf. Humanoid Robots, Humanoids, 2005, pp. 442–449.
[18] N. Pollard, Synthesizing grasps from generalized prototypes, in: Proc. IEEE Int. Conf. Robotics and Automation, ICRA, 1996, pp. 2124–2130.
[19] K. Hsiao, T. Lozano-Pérez, Imitation learning of whole-body grasps, in: Proc. IEEE Conf. Intelligent Robots and Systems, IROS, 2006, pp. 5657–5662.
[20] A. Saxena, J. Driemeyer, J. Kearns, A. Ng, Robotic grasping of novel objects, in: Advances in Neural Information Processing Systems, NIPS, 2006.
[21] M. Ciocarlie, C. Goldfeder, P. Allen, Dimensionality reduction for hand-independent dexterous robotic grasping, in: Proc. IEEE Conf. Intelligent Robots and Systems, IROS, 2007, pp. 3270–3275.
[22] L.Y. Chang, G. Zeglin, N. Pollard, Preparatory object rotation as a human-inspired grasping strategy, in: Proc. IEEE Int. Conf. Humanoid Robots, Humanoids, 2008, pp. 527–534.

[23] K. Hsiao, L.P. Kaelbling, T. Lozano-Pérez, Task-driven tactile exploration, in: Proceedings of Robotics: Science and Systems, RSS, 2010.
[24] L.Y. Chang, S. Srinivasa, N. Pollard, Planning pre-grasp manipulation for transport tasks, in: Proc. IEEE Int. Conf. Robotics and Automation, ICRA, 2010, pp. 2697–2704.
[25] M.T. Mason, Mechanics of Robotic Manipulation, MIT Press, Cambridge, MA, 2001.
[26] T. Lozano-Perez, M. Mason, R. Taylor, Automatic synthesis of fine-motion strategies for robots, The International Journal of Robotics Research (1984) 13–24.
[27] K. Lynch, M. Mason, Stable pushing: mechanics, controllability, and planning, The International Journal of Robotics Research (1996).
[28] M. Dogar, S. Srinivasa, Push-grasping with dexterous hands: Mechanics and a method, in: Proc. IEEE Conf. Intelligent Robots and Systems, IROS, 2010, pp. 2123–2130.
[29] K. Lynch, Toppling manipulation, in: Proc. IEEE Int. Conf. Robotics and Automation, ICRA, 1999, pp. 2551–2557.
[30] K. Hauser, V. Ng-Thow-Hing, Randomized multi-modal motion planning for a humanoid robot manipulation task, The International Journal of Robotics Research (2010).
[31] E. Yoshida, M. Poirier, J.-P. Laumond, O. Kanoun, F. Lamiraux, R. Alami, K. Yokoi, Pivoting based manipulation by a humanoid robot, Autonomous Robots (2010) 77–88.
[32] D. Berenson, R. Diankov, K. Nishiwaki, S. Kagami, J. Kuffner, Grasp planning in complex scenes, in: Proc. IEEE Int. Conf. Humanoid Robots, Humanoids, 2007, pp. 42–48.
[33] M. Przybylski, T. Asfour, R. Dillmann, Unions of balls for shape approximation in robot grasping, in: Proc. IEEE Conf. Intelligent Robots and Systems, IROS, 2010, pp. 1592–1599.
[34] A. Bicchi, V. Kumar, Robotic grasping and contact: a review, in: Proc. IEEE Int. Conf. Robotics and Automation, ICRA, 2000, pp. 348–353.
[35] C. Ferrari, J. Canny, Planning optimal grasps, in: Proc. IEEE Int. Conf. Robotics and Automation, ICRA, 1992, pp. 2290–2295.
[36] A.T. Miller, P.K. Allen, Examples of 3d grasp quality computations, in: Proc. IEEE Int. Conf. Robotics and Automation, ICRA, vol. 2, 1999.
[37] D. Bertram, J. Kuffner, R. Dillmann, T. Asfour, An integrated approach to inverse kinematics and path planning for redundant manipulators, in: Proc. IEEE Int. Conf. Robotics and Automation, ICRA, 2006, pp. 1874–1879.
[38] S.R. Buss, Introduction to inverse kinematics with Jacobian transpose, pseudoinverse and damped least squares methods, ucsd.edu/~sbuss/ResearchWeb, 2004.
[39] D. Tolani, A. Goswami, N. Badler, Real-time inverse kinematics techniques for anthropomorphic limbs, Graphical Models (2000) 353–388.
[40] J. Latombe, Robot Motion Planning, Springer, 1991.
[41] S. LaValle, Rapidly-exploring random trees: A new tool for path planning, Tech. Rep., TR 98-11, Computer Science Dept., Iowa State University, 1998.
[42] L.Y. Chang, N.S. Pollard, Video survey of pre-grasp interactions in natural hand activities, in: Robotics: Science and Systems, RSS 2009 Workshop: Understanding the Human Hand for Advancing Robotic Manipulation, 2009.
[43] L.Y. Chang, R.L. Klatzky, N.S. Pollard, Selection criteria for preparatory object rotation in manual lifting actions, Journal of Motor Behavior (2010) 11–27.
[44] J. Kim, Example-based grasp adaptation, Master's Thesis, Dept. of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, 2007.
[45] R. Diankov, J. Kuffner, Openrave: a planning architecture for autonomous robotics, Tech. Rep. CMU-RI-TR-08-34, Robotics Institute, 2008.
[46] M. Dogar, S. Srinivasa, Push-grasping with dexterous hands: Mechanics and a method, in: Proc. IEEE Conf. Intelligent Robots and Systems, IROS, 2010.

Daniel Kappler is currently a Diploma student in Computer Science at Karlsruhe Institute of Technology, Karlsruhe, Germany. His major research interests include biological inspired machine learning especially in the research area of humanoid robotics, dexterous manipulation, and grasping.

Lillian Y. Chang is a Computing Innovation Fellow with Intel Corporation and a Research Associate at the University of Washington. She completed her Ph.D. in Robotics at the Robotics Institute in the School of Computer Science at Carnegie Mellon University. Her research interests include manipulation and motion planning, quality-of-life applications of robotics, and interfaces for human–robot interaction. Lillian is a former NASA Harriett G. Jenkins Pre-doctoral Fellow and former NSF Graduate Research Fellow.

Nancy S. Pollard is an Associate Professor in the Robotics Institute and the Computer Science Department at Carnegie Mellon University. She received her Ph.D. in Electrical Engineering and Computer Science from the MIT Artificial Intelligence Laboratory in 1994, where she performed research on grasp planning for articulated robot hands. Before joining CMU, Nancy was an Assistant Professor and part of the Computer Graphics Group at Brown University. Her primary research objective is to understand how to create natural motion for animated human characters and humanoid robots.

Tamim Asfour is a senior research scientist and leader of the Humanoid Research Group at Humanoids and Intelligence Systems Lab, Institute for Anthropomatics, Karlsruhe Institute of Technology (KIT). In 2003 he was awarded with the Research Center for Information Technology (FZI) price for his outstanding Ph.D. Thesis on sensorimotor control in humanoid robotics and the development of the humanoid robot ARMAR. His major research interest is humanoid robotics.

Rüdiger Dillmann is full professor at the Karlsruhe Institute of Technology (KIT), Department of Informatics, and head of the Humanoids and Intelligence Systems Laboratory at the Institute for Anthropomatics. His major research interests include humanoid robotics and human-centered robotics, robot programming by demonstration, machine learning, robot vision, cognitive cars, service robots, and medical applications of informatics.