

Unified Representation of Pre-Grasp Strategies for Object Manipulation

Daniel Kappler*, Lillian Chang[†], Markus Przybylski*, Nancy Pollard[†], Tamim Asfour*, and Rüdiger Dillmann*

*Institute for Anthropomatics, Karlsruhe Institute of Technology, Karlsruhe, Germany.

[†]Robotics Institute, School of Computer Science, Carnegie Mellon University, Pittsburgh, Pennsylvania, United States.

*daniel.kappler@student.kit.edu, {markus.przybylski, asfour, dillmann}@kit.edu,

[†]{lillianc@ri, nsp@cs}.cmu.edu

Abstract—In this paper, we present a method for representing and re-targeting manipulations for object adjustment before final grasping. Such *pre-grasp manipulation actions* bring objects into better configurations for grasping through e.g. object rotation or object sliding. For this purpose, we propose a scaling-invariant and rotation-invariant representation of the hand poses, which is then automatically adapted to the target object to perform the selected pre-grasp manipulations. We show that pre-grasp strategies such as sliding manipulations not only enable more robust object grasping, but also significantly increase the success rate for grasping.

I. INTRODUCTION

In recent years, many humanoid robots like ARMAR [1], Justin [2], HRP2 [3], Domo [4] and Twendy-One [5] have been designed to assist humans. The major tasks people expect of robots are helping in a kitchen environment, cleaning a room, and moving heavy objects [6]. For all these tasks, robust fetching of various objects from a surface is a fundamental subtask. In this paper, we propose a pre-grasp manipulation strategy to allow changing objects to a better configuration before final grasping. This strategy is inspired by human-object interaction strategies observed in grasping tasks. Examples 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 [7].

Motivated by the study about human pre-grasp strategies based on preparatory object rotation introduced by Chang et al. in [7], we conducted an informal video survey to investigate possible object-based patterns in everyday human-object interaction. In our study, participants grasped and moved household objects such as a stapler, books, and CDs that were scattered on a table. We observed that people tend to pre-manipulate objects, e.g. slide objects on the table if they are out of reach or inconvenient to grasp. Furthermore, they used similar hand poses for pre-grasp manipulation when grasping objects of similar shapes.

The work described in this paper was 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). This work was also supported by the National Science Foundation (CCF-0702443). L. Y. Chang received support from a NASA Harriet G. Jenkins Pre-Doctoral Fellowship.

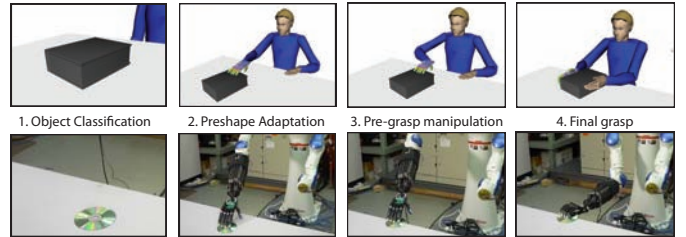


Fig. 1. Overview of the pre-grasp strategy, where an object is slid on the table before the final grasp.

Based on these observed patterns, we propose a data-driven method which enables automatic synthesis of pre-grasp manipulation strategies for various objects. To facilitate manipulation planning, we propose to learn and store preshapes and pre-grasp manipulations according to object categories. A preshape in this context is a data structure which combines starting hand poses and configurations, constraints on the related pre-grasp manipulation actions as well as suitable final grasp-types. Pre-grasp manipulation actions, such as sliding, are annotated with context information in our pre-grasp manipulation data structure which contains object and robot constraints for manipulation, and a final region within which the action should relocate the object.

Current methods [8] [9] [10] 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 a 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.

II. RELATED WORK

Our approach to use human manipulation strategies to plan object grasping is inspired by the work of Chang et al. [7], [11], where they present an augmented grasp strategy based on pre-grasp rotation for objects with handles (such as a cooking pan) to increase the success rate for finding stable lifting grasps.

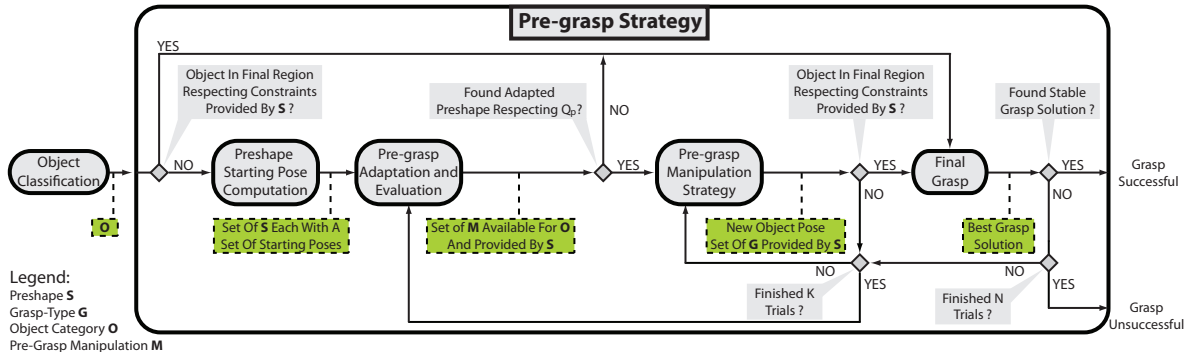


Fig. 2. The proposed pre-grasp strategy architecture with the four phases: “Preshape Starting Pose Computation, Pre-grasp Adaptation and Evaluation, Pre-grasp Manipulation, Final Grasp” embedded in the grasp planning context.

Our presented pre-grasp strategy builds upon multiple concepts of motion planning. One necessary feature for the strategy is to determine if specific hand poses (position and orientation of the palm) and configuration (finger joint values) are reachable or not. Many humanoid robots have at least 7-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 [13]–[15]. 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 [16] discusses traditional approaches to this problem and Rapidly-exploring Random Tree (RRT) methods by Lavelle [17]. For pre-grasp manipulation, there are many methods to adjust the object on the support surface. Pushing and sliding, which are used in this paper, are discussed in [18]–[20] and reorientation was used in [7], [12], [33] to successfully grasp hard-to-reach objects. Toppling as well as tumbling presented by Lynch et al. [21] are also possible pre-grasp manipulations.

The key problem for fetching objects from a surface is finding a stable grasp to lift the object. Thus, many past approaches focused on automatically generating hand poses for stable grasping. Saxena et al. [22] proposed a method based on 2D image analysis. The majority of grasping research works with 3D object models. Goldfeder et al. [23] investigated grasp pose adaptation based on the decomposition of the object trimesh into superquadrics. Using template grasp poses for grasping in general was proposed by Pollard [24], Miller et al. [8] and Hsiao [25]. Berenson et al. [9] proposed a method to precompute a set of possible grasps offline for later online evaluation against environmental constraints. Przybylski et al. suggested in [10] 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 [26]. For evaluating the stability of a possible lifting grasp, Ferrari and Canny [27] 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 [28].

III. 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 to adjust the object to a final pose, within a distinct final region, using a suitable pre-grasp manipulation strategy. 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 which are described in the following sections.

A. Representations

At the high level, the context knowledge is organized by *object categories* \mathbb{O} . 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, a set of *preshapes* \mathbb{S} and a set of *pre-grasp manipulation* data structures \mathbb{M} .

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

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

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

$$\mathbb{S} = (\mathbf{c}, P, C, \mathbf{G}, \mathbb{M}) \quad (2)$$

Every preshape provides a hand configuration \mathbf{c} , which is defined by the joint values of the given robot hand. 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. 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 , 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.

2) *Pre-grasp manipulation*: The goal for 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 \mathbb{M} data structure:

$$\mathbb{M} = (a, C, F) \quad (3)$$

A pre-grasp manipulation is defined by an action a , which in this context may be for example toppling, tumbling, rotating, pushing, and sliding. In general an action a has to satisfy constraints C during manipulation, for example constant object contact, force limitations to the robot joints as well as to the object surface, or object orientations like ensuring that a cup is not spilling. Additionally for particular grasp-types \mathbf{G} and object categories \mathbb{O} there are constraints C 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, which allows us to select preshapes \mathbb{S} and pre-grasp manipulations \mathbb{M} based on the object category \mathbb{O} . These data structures build the foundation for general usage of pre-grasp strategies to increase the success rate of stable grasp acquisition.

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 I 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 quantities in the object coordinate system, illustrated in Fig. 3:

- The initial position \mathbf{x}_s on the object surface,
- the offset \mathbf{f} of the hand to the object surface,
- the dimension \mathbf{d} of the original object's bounding box,

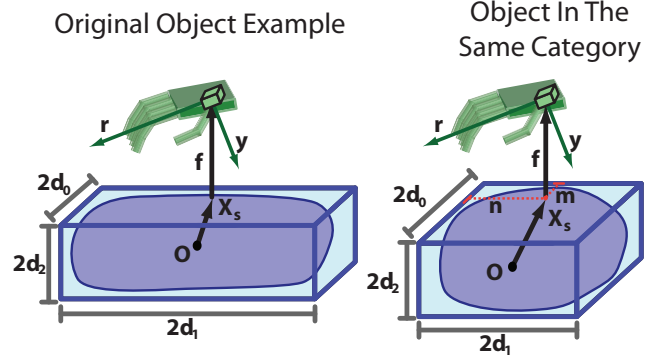


Fig. 3. Representation of stored information in the preshape \mathbb{S} structure. These components allow for automatic determination of the starting hand poses relative to the object. The left side shows the original object from the preshape example. The right side shows one possible starting pose made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

- and the hand orientation stored by the roll axis \mathbf{r} and yaw axis \mathbf{y}

The separation of the starting poses into the 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 III-C. The assigned pre-grasp manipulation \mathbb{M} is always sliding manipulation. Constraints C in our context are fingertip contact with the object and collision free arm configuration for the pose. 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.

Sliding is the pre-grasp manipulation action a used in this paper. Constraints C for this \mathbb{M} are that the fingers are in contact during the whole sliding manipulation, and continuous arm configurations for the whole path can be found. Object pose constraints are not considered due to the fact that no special grasps for objects such as handled ones are available.

B. 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 procedure is visualized in Fig. 2. The method takes an object category \mathbb{O} described in detail in Section III-A as input which can be gathered through a classification step to find the best matching \mathbb{O} for a given object.

At 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 step kinematic template preshapes of \mathbb{O} are adapted to the object surface. The *adapted preshapes* are *evaluated* with a rating function such as $Q_p(\mathbf{c}, \mathbf{p}, \mathbf{q}, F, C)$ 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 IV-B checks finger contact with the object, the arm pose, the size of the final region F , and the compliance with constraints.

The next step performs a *corresponding pre-grasp manipulation* M to the best rated S . A successful pre-grasp manipulation plan is achieved if the object is within the final region F satisfying the constraints C . If no successful plan is found the best manipulation regarding $Q_m(\mathbf{p}_f, F, C, g)$ is performed and the search is restarted at the previous step. Q_m described in Section IV-B determines which manipulation will be performed based on the object pose, the distance to the final region, the observed constraints, and the grasp success.

Finally, the grasp-types G usable for a object pose \mathbf{p}_f are *selected and adapted* to the object. If no successful solution respecting 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 for a certain amount of trials another pre-grasp manipulation available by the adapted preshape is chosen to successfully find a final grasp.

In the following Sections III-C to III-F we describe the four individual parts of the proposed pre-grasp strategy: start pose computation, pose adaptation, pre-grasp manipulation, and final grasping.

C. Preshape Starting Pose Computation

As described in Section III-A every preshape has to have the ability to serve a set of starting poses P . 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 III-A. 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. 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{y} . 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 regardless of the object pose we transform the object coordinate system so that the robot shoulder position is expressed by a positive vector.

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. The absolute and relative distance either does not or does change with scaling, respectively. The absolute relative 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, 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.

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.

D. Preshape Adaptation and Evaluation

The second phase, preshape adaptation (Fig. 2), builds upon the set of preshapes selected by the object category and P determined by the previous step. The key aspect is that preshapes 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 Hsiao [25] proposed a solution for template grasp adaptation for direct grasping using starting position mapping on related objects. Kim [29] suggested a grasp adaption algorithm based on mapping grasp positions from an example to a new object. Another grasp adaptation method from generic prototypes is discussed by Pollard [24].

Since we need a grasp position for pre-grasp manipulation and not only for direct grasping, we present a new adaptation algorithm 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. The starting poses are checked for reachability to generate collision-free arm IK solutions. 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. The adaptation process has two parts, one for adapting hand configurations – changing the hand configuration \mathbf{c} to get fingertip contact, and another one to relocate the hand if no solution regarding Q_p can be determined (Fig. 4).

Now we describe the fingertip contact calculation. Due to the configuration \mathbf{c} given by preshape 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 the initial finger configuration is used if no contact solution can be found. This method is described in detail in Fig. 4. Once the preshapes have been adapted to the object, we then compare them to select the best adapted preshape for the pre-grasp manipulation phase. The adapted preshapes are scored by a rating function $Q_p(\mathbf{c}, \mathbf{p}, \mathbf{q}, F, C)$ regarding the pre-grasp manipulations they correspond to and the best is propagated to the pre-grasp manipulation phase.

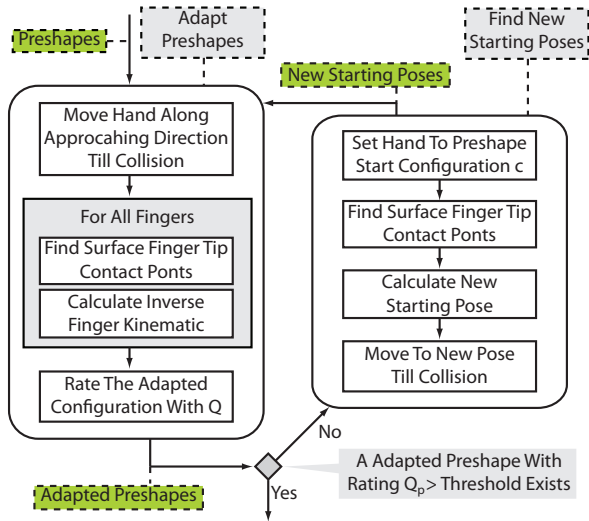


Fig. 4. The left side of our proposed algorithm adapts a hand configuration of a given preshape to the object surface. If no adapted preshape with a high enough rating Q_p is available, the right side of the algorithm computes new starting poses.

Although the initial hand pose $\mathbf{p} \in P$ is likely to be in a promising spot for finding a successful solution with high value of Q_p , it may occur that no finger contacts can be found. If no preshape is successfully adapted, a new P has to be found to achieve successful solutions with respect to Q_p . In these cases, we use the original preshape configuration \mathbf{c} , and then we search for a new hand pose \mathbf{p} based on the closest surface points to the active finger tips. After the hand is iteratively moved to the new pose, the process of finding fingertip contact restarts (Fig. 4).

E. Pre-grasp Manipulation

Currently, the pre-grasp strategy has achieved the following goals. A set of preshapes have been adapted to the object and the best adapted preshape relative to a quality function Q_p has been selected. Only pre-grasp manipulations related to the object category and adapted preshape are available at this stage. In addition to that the adapted preshape has to respect the constraints for the pre-grasp manipulation.

The goal of the pre-grasp manipulation phase is to find a path such that the object is relocated to the final region and respects the pose constraint of the selected preshape, manipulation, and grasp. The kind of pre-grasp manipulation used for readjusting the object location is specified within the selected and adapted preshape \mathbb{S} and the selected object category \mathbb{O} .

Pre-grasp strategies in general have two aspects in common. First, they use the benefit that the object rests on a surface, such that it is often it is easier to adjust the object pose than to grasp the object. Second, they solve the problem of finding successful and stable grasps if the object is not within a certain reachable area or pose by relocating the object to more easier plan lifting grasps.

The general idea of this paper to provide a representation that enables to use multiple pre-grasp manipulations such as

sliding manipulations proposed by Lynch et al. [20], rotating manipulations discussed by Chang et al. [7], for a pre-grasp strategy to adjust the object prior to grasping.

With completion of this phase, the object is located within the final region and is satisfying the pose constraints, and thus the grasping action can be planned more easily.

F. 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 . Finding grasp candidates for an object located in such a region has been well explored in robotics and computer graphics. Literature such as [8] [9] [10] [24] [25] about this topic was introduced in Section II. If a successful grasp solution respecting force closure metric is found the pre-grasp strategy is finished. But if no stable solution can be found (Fig. 2), either the object has to be relocated based on the previous step, or the manipulation strategy has to be changed.

IV. IMPLEMENTATION

We implemented our framework as a plugin in OpenRAVE [30] which provides collision checking, IK solutions and 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. 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.

A. Object Classification

Mapping an input feature vector, e.g. weight and dimension of a given object, to an output label, e.g. box or cylinder, is a standard classification problem. There are many different solutions to solve this problem, for example neural networks [31] or support vector machines [32]. It is possible to use offline, online, supervised or unsupervised learning in our framework, which allows for flexible implementations.

In our implementation we use the dimensions, weight and curvature as input vector for a multilayer neural network. The dimensions and curvature are computed online based on the object trimesh. This classification results in assigning an object category. 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 III-D. The object category is described in detail in Section III-A.

B. Rating Function

As introduced in Section III-D the adapted pre-shapes are evaluated by the corresponding rating function $Q_p(\mathbf{c}, \mathbf{p}, \mathbf{q}, F, C)$. 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:

$$Q_p(\mathbf{c}, \mathbf{p}, \mathbf{q}, F, C) = \alpha_1 \sum_{i=1}^n b_i + \alpha_2 \text{size}(F) + \alpha_3 \text{sat}(C) + \alpha_4 \sum_{i=1}^a \left(\frac{(\min_i - q_i)^2 + (\max_x - q_i)^2}{(\min_i - \max_x)^2} \right)^{-1} \quad (4)$$

where n is the number of fingers, a the number of arm joints, b_i is a binary indicator expressing whether the finger i made contact. The relative size of the final region \mathbb{F} is determined by $\text{size}(\mathbb{F})$. The constraints \mathbb{C} are checked to determine if a related pre-grasp manipulation can be performed, for example if the force on the object is sufficiently away from the limit force. The joint limits are \min_i and \max_x , the current joint value is q_i . The different parts of the equation are weighted by α_i .

A second rating function Q_m introduced in Section III-E is designed to determine the pose \mathbf{p}_f for unsuccessful pre-grasp manipulation which is the best starting pose for a new adaptation set.

$$Q_m(\mathbf{p}_f, F, C, g) = \alpha_1 d(\mathbf{p}_f, F) + \alpha_2 d(\mathbf{p}_f, C) + \alpha_2 g \quad (5)$$

where the distance to the final region F is measured as well as to object pose constraints. In addition if the object was within the final region and not graspable determined by g , the final region for this grasp-type has to be removed for later trials.

C. Pre-grasp Manipulation

As mentioned in Section III-A sliding is the only pre-grasp manipulation implemented for evaluation in this paper. This is based on the observation within the video survey mentioned in Section I where sliding manipulation has been the most common pre-grasp manipulation strategy for the given objects. For other pre-grasp strategies like pre-grasp rotation we refer to the work of Chang et al. [7], [11], [33]; pre-grasp toppling and tumbling to Lynch et al. [21]; sliding by Dogar et al. [34].

For a successful sliding manipulation we try to find a continuous trajectory from the initial adapted preshape pose to a manipulation end pose so that the object is within the final region without losing the hand/object and the object/surface contacts. For sliding manipulation we have no pose constraint within the final region due to the fact that the manipulating hand is already in contact with the object during the whole manipulation. Due to the definition of the final region (Section III-A) it is assumed that after successful pre-grasp manipulation the object is more likely to be graspable after successful pre-grasp manipulation.

When the hand is in contact with the object during sliding pre-grasp manipulation, we require that the finger object

friction is high enough such that the object moves along with the hand movement on the surface, which is assumed to be true as long as contact between the finger and the object is maintained.

D. 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. [9] propose 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} we sample the object surface and adapt the provided template grasps to the object surface with the algorithm described in Fig. 4.

After grasp candidates are found the stability has to be determined. A stable solution in this context has to satisfy the force-closure metric proposed by Ferrari et al. [27].

V. 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.

A. 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 [30] with 7-DoF arms and 15-DoF hands (Fig. 1). In our implementation, the right hand has been replaced with the ShadowHand with 23 DoFs.

Two objects from each of the four sliding manipulation preshape sets were tested (Table I). 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 provide 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 20s.

Each object is placed at a random position and plane orientation on the table within a 1.2m×0.6m region in front of the robot, shifted 0.4 m right of the robot center. We discard the random position if the object is not within the reaching radius of the robot's right arm. We generate poses in this manner to obtain 15 reachable poses per object. Both planners attempt an object acquisition solution for the same 15 starting object poses.

B. Direct Grasp Planner

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

TABLE I
OBJECTS AND OBJECT CATEGORIES*.

\mathbb{O}_0 flat shape	\mathbb{O}_1 lightweight box	\mathbb{O}_2 heavy box	\mathbb{O}_3 cylinder
CD	bat	book	tin
credit card	stapler	keyboard box	sugar canister
15-cm ruler	tape dispenser	VGA-splitter	plant
house key	cassette tape	sugar box	water jug
	bed linen box		
	portable hard-drive		

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

C. Simulation Results

Table II 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 Pose” 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 \mathbb{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 increases the success rate for object acquisition for all tests as shown in Table II. In addition, our approach reduces the computation time for grasp adaptation significantly regardless of whether the object is directly graspable or not. Furthermore median of the planning time for the grasp pose with our method was about 2s whereas the direct grasp planned took about 200s. The long computation time for the direct grasp planner results through the exhaustive and in the most cases not successful search for stable final grasp candidates.

D. 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 grasping. Participants selected the preferred video in each pair. Table III shows that pre-grasp rotation 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 ($p(X^2 = 5.76, df = 1) = 0.02$).

E. Physical Demonstration

We demonstrated the physical plausibility of our simulated pre-grasp strategy plans on a multi-fingered robot manipulator. The system consists of a 7-DoF Motoman arm and an

TABLE II
SIMULATION RESULTS FOR THE METHOD COMPARISON.

	Successes out of 30	Mean planning times (seconds)		
		Pre-grasp	Grasp	Trajectory
CD, ruler				
Pre-grasp push	25	25.1	42.8	53.9
Direct grasp	1	–	259.4	24.1
bat, stapler				
Pre-grasp push	24	2.0	78.5	64.1
Direct grasp	21	–	99.7	60.2
book, sugar box				
Pre-grasp push	29	37.0	42.7	66.6
Direct grasp	18	–	148.5	45.9
tin, water jug				
Pre-grasp push	30	8.6	9.5	72.9
Direct grasp	19	–	118.7	48.3
Total	out of 120	Pre-grasp	Grasp	Trajectory
Pre-grasp push	108	18.2	43.4	64.4
Direct grasp	59	–	156.6	44.6

TABLE III
THE NUMBER OF PARTICIPANTS WHO PREFERRED EITHER PRE-GRASP INTERACTION OR DIRECT GRASPING IN THE VIDEO SURVEY.

Object	Manipulation method	
	Direct grasp	Pre-grasp push
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

attached Shadowhand robot with 5 fingers. In our example demonstration, the object is a CD, which is difficult to grasp from a table because of its thin edge. However, the Motoman with Shadowhand 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 Shadowhand were selected from our simulated plan but were manually pre-set to match the arm trajectory timing. The video at <http://his.anthropomatik.kit.edu/english/532.php> shows the comparison between the simulated Motoman plan and the physical execution for the CD object.

VI. DISCUSSION AND CONCLUSION

In this paper we introduced a representation framework of pre-grasp strategies for object manipulation allowing the use of human manipulation strategies for object grasping. We proposed a new object representation based on preshapes and pre-grasp manipulations which provides starting hand

configuration and poses suitable for adaptation to the object surface, regardless to object rotation and scaling. With this information, pre-grasp manipulation actions, which includes 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.

The most important next steps are, to include more pre-grasp manipulations and preshape data structure to support the thesis and to update the sliding one to a more sophisticated version which is capable of avoiding obstacles on the 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.

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 [9] 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.

Overall, the proposed unified 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.

VII. ACKNOWLEDGMENTS

We thank the InterACT program for making this joint research project possible. The authors thank Justin Macey for help with the motion capture, Sharon R. Hoosein and James Chan for object modeling, Garth Zeglin for help with the Motoman robot, Yue Peng Toh for help with the video survey, and Rosen Diankov for the OpenRAVE support.

REFERENCES

- [1] T. Asfour, K. Regenstein, P. Azad, J. Schroder, A. Bierbaum, N. Vahrenkamp, and 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, and G. Hirzinger, "Impedance Behaviors for Two-handed Manipulation: Design and Experiments," *Proc. IEEE Int. Conf. Robots Automation (ICRA)*, pp. 4182–4189, 2007.
- [3] K. Okada, T. Ogura, A. Haneda, J. Fujimoto, F. Gravot, and M. Inaba, "Humanoid motion generation system on HRP2-JSK for daily life environment," *IEEE International Conference Mechatronics and Automation*, pp. 1772–1777, 2005.
- [4] A. Edsinger-gonzalez and J. Webber, "Domo: a force sensing humanoid robot for manipulation research," *Proc. IEEE Int. Conf. Humanoid Robots (Humanoids)*, 2004.
- [5] H. Iwata and S. Sugano, "Design of human symbiotic robot twenty-one," in *Proc. IEEE Int. Conf. Robots Automation (ICRA)*, 2009.
- [6] C. Ray, F. Mondada, and R. Siegwart, "What do people expect from robots?" in *Proc. IEEE Conf. Intelligent Robots and Systems (IROS)*, 2008, pp. 3816–3821.
- [7] L. Y. Chang, G. Zeglin, and N. Pollard, "Preparatory object rotation as a human-inspired grasping strategy," in *Proc. IEEE Int. Conf. Humanoid Robots (Humanoids)*, 2008, pp. 527–534.
- [8] A. Miller, S. Knoop, H. Christensen, and P. Allen, "Automatic grasp planning using shape primitives," in *Proc. IEEE Int. Conf. Robots Automation (ICRA)*, vol. 2, 2003, pp. 1824–1829.
- [9] D. Berenson, R. Diankov, K. Nishiwaki, S. Kagami, and J. Kuffner, "Grasp planning in complex scenes," in *Proc. IEEE Int. Conf. Humanoid Robots (Humanoids)*, 2007, pp. 42–48.
- [10] M. Przybylski, T. Asfour, and R. Dillmann, "Unions of balls for shape approximation in robot grasping," in *To appear in: Proc. IEEE Conf. Intelligent Robots and Systems (IROS)*, 2010.
- [11] L. Chang, R. Klatzky, and N. Pollard, "Selection Criteria for Preparatory Object Rotation in Manual Lifting Actions," *Journal of Motor Behavior*, pp. 11–27, 2009.
- [12] K. Hsiao, L. Kaelbling, and T. Lozano-Pérez, "Task-Driven Tactile Exploration," 2010.
- [13] D. Bertram, J. Kuffner, R. Dillmann, and T. Asfour, "An integrated approach to inverse kinematics and path planning for redundant manipulators," *Proc. IEEE Int. Conf. Robots Automation (ICRA)*, pp. 1874–1879, 2006.
- [14] S. R. Buss, "Introduction to inverse kinematics with Jacobian transpose, pseudoinverse and damped least squares methods." ucsd.edu/~sbuss/ResearchWeb, 2004.
- [15] D. Tolani, A. Goswami, and N. Badler, "Real-time inverse kinematics techniques for anthropomorphic limbs," *Graphical models*, pp. 353–388, 2000.
- [16] J. Latombe, *Robot motion planning*. Springer, 1991.
- [17] S. LaValle, "Rapidly-exploring random trees: A new tool for path planning," TR 98-11, Computer Science Dept., Iowa State University, Tech. Rep., 1998.
- [18] M. T. Mason, "Mechanics of Robotic Manipulation," Cambridge, MA: MIT Press, 2001.
- [19] T. Lozano-Perez, M. Mason, and R. Taylor, "Automatic synthesis of fine-motion strategies for robots," *The International Journal of Robotics Research*, pp. 13–24, 1984.
- [20] K. Lynch and M. Mason, "Stable pushing: Mechanics, controllability, and planning," *The International Journal of Robotics Research*, 1996.
- [21] K. Lynch, "Toppling manipulation," in *Proc. IEEE Int. Conf. Robots Automation (ICRA)*, 1999, pp. 2551–2557.
- [22] A. Saxena, J. Driemeyer, J. Kearns, and A. Ng, "Robotic grasping of novel objects," *Advances in Neural Information Processing Systems (NIPS)*, 2007.
- [23] C. Goldfeder, P. Allen, C. Lackner, and R. Pelosof, "Grasp planning via decomposition trees," in *Proc. IEEE Int. Conf. Robots Automation (ICRA)*, 2007, pp. 10–14.
- [24] N. Pollard, "Synthesizing grasps from generalized prototypes," in *Proc. IEEE Int. Conf. Robots Automation (ICRA)*, 1996, pp. 2124–2130.
- [25] K. Hsiao and T. Lozano-Perez, "Imitation learning of whole-body grasps," in *Proc. IEEE Conf. Intelligent Robots and Systems (IROS)*, 2006, pp. 5657–5662.
- [26] A. Bicchi and V. Kumar, "Robotic grasping and contact: a review," in *Proc. IEEE Int. Conf. Robots Automation (ICRA)*, 2000, pp. 348–353.
- [27] C. Ferrari and J. Canny, "Planning optimal grasps," in *Proc. IEEE Int. Conf. Robots Automation (ICRA)*, 1992, pp. 2290–2295.
- [28] A. T. Miller and P. K. Allen, "Examples of 3D Grasp Quality Computations," *Proc. IEEE Int. Conf. Robots Automation (ICRA)*, vol. 2, 1999.
- [29] J. Kim, "Example-based grasp adaptation," Master's thesis, Dept. of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, 2007.
- [30] R. Diankov and J. Kuffner, "Openrave: A planning architecture for autonomous robotics," Robotics Institute, Tech. Rep. CMU-RI-TR-08-34, 2008.
- [31] S. Haykin, *Neural networks: a comprehensive foundation*. Prentice Hall, 2008.
- [32] C. Burges, "A tutorial on support vector machines for pattern recognition," *Data mining and knowledge discovery*, pp. 121–167, 1998.
- [33] L. Y. Chang, S. Srinivasa, and N. Pollard, "Planning pre-grasp manipulation for transport tasks," in *Proc. IEEE Int. Conf. Robots Automation (ICRA)*, May 2010.
- [34] M. Dogar and S. Srinivasa, "Push-grasping with dexterous hands: Mechanics and a method," in *Proc. IEEE Conf. Intelligent Robots and Systems (IROS)*, October 2010.