

Robust Object Grasping using Force Compliant Motion Primitives

Moslem Kazemi
Jean-Sebastien Valois
J. Andrew Bagnell
Nancy Pollard

CMU-RI-TR-12-04

January 30, 2012

Robotics Institute
Carnegie Mellon University
Pittsburgh, Pennsylvania 15213

© Carnegie Mellon University

Abstract

We address the problem of grasping everyday objects that are small relative to an anthropomorphic hand, such as pens, screwdrivers, cellphones, and hammers from their natural poses on a support surface, e.g., a table top. In such conditions, state of the art grasp generation techniques fail to provide robust, achievable solutions due to either ignoring or trying to avoid contact with the support surface. In contrast, we show that contact with support surfaces is critical for grasping small objects. This also conforms with our anecdotal observations of human grasping behaviors. We develop a simple closed-loop hybrid controller that mimics this interactive, contact-rich strategy by a position-force, pre-grasp and landing strategy for finger placement. The approach uses a compliant control of the hand during the grasp and release of objects in order to preserve safety. We conducted extensive grasping experiments on a variety of small objects with similar shape and size. The results demonstrate that our approach is robust to localization uncertainties and applies to many real-world objects.

Contents

1	Introduction	1
2	Related Work	2
3	Hardware and Control Requirements	3
3.1	Hardware Requirements	3
3.2	Low-Level Controllers for Hand and Fingers	4
3.2.1	Velocity-based Operational Space Control of Hand	4
3.2.2	Position-based Control of Fingers	4
4	Force Compliant Grasping Primitives	5
4.1	Force Compliant Fingers Landing	5
4.2	Force Compliant Grasping	7
4.3	Force Compliant Object Release	8
5	Experiments and Results	9
5.1	Fingers Landing Experiments	10
5.2	Object Grasping Experiments	12
5.3	Object Release Experiments	14
5.4	Parallel with Human Grasping	16
6	Conclusion	17

1 Introduction

Grasping constitutes an essential component in an autonomous robotic manipulation system operating in human environments. The wide variety of everyday objects and various environment settings requires grasping strategies that are robust to variations in shape, size, and pose, as well as uncertainties in perception and robot kinematics. Although advances in perception provide accurate object pose estimation, calibration and kinematic factors affect the accuracy at which an end-effector can be controlled for grasping. Because of such uncertainties contact information such as tactile and force/torque feedback is necessary to achieve robust grasps.

Much research in robotic manipulation is focused on geometric grasp generation and planning. Researchers study finger and object interactions and develop grasp quality metrics based on form and force closure [1]. Despite promising progress in geometric grasp planning and the efforts devoted to analyzing grasp properties, challenges still remain in real-world manipulation tasks because these fails to consider opportunities presented by contacts with the environment. In our view, supported by our empirical studies, two factors play a crucial role in achieving more robust grasps: dealing with positioning uncertainties [5], and using compliant motions [10] to handle contacts between the robot and the support surface, or the object.

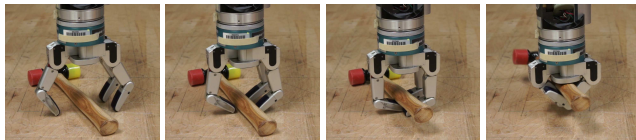


Figure 1: Grasping of a hammer composed of three steps: compliant landing, compliant grasping, and compliant releasing

Our work (Fig.1) is motivated by the task of grasping everyday objects that are small relative to an anthropomorphic hand, including a pen, screwdriver, cellphone, and hammer from their natural poses on a support surface, e.g., a table top (see Fig.2). Note that existing techniques for grasping larger objects generally tend to perform well when reliable grasp points can be found away from the support surfaces. However, that is rarely the case for small objects. In addition, because existing grasp planning tools (such as OpenRAVE [3] and GraspIt![11]) rely on precise finger-to-object contact points while avoiding the surrounding environment they are impractical for grasping small objects. Figure 3 illustrates some of the failure modes of a geometric grasp planner.

In our experience, geometric techniques are only effective when precise calibration is performed to position the robot relative to the object. In practice, limitations in sensing and control uncertainty dominates and makes it impractical. Thus, it is important to devise grasping strategies that are robust to these errors.

Most approaches to grasping attempt to model uncertainty. In contrast, we believe that the effect of uncertainty can be ignored by compliant interactions between the robot and the environment. Because we allow contact with the environment, compliant

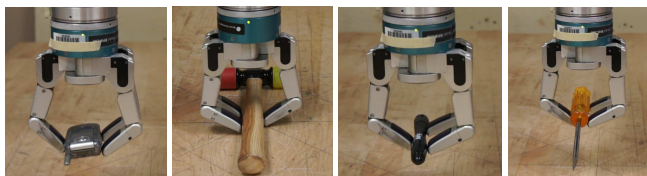


Figure 2: The anthropomorphic Barrett hand grasps small objects: a cellphone, hammer, pen, and screwdriver from a table top.

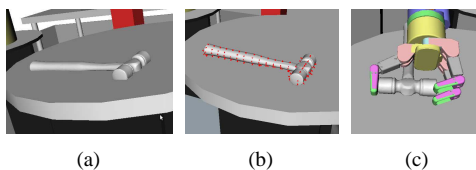


Figure 3: Failure modes of a geometric grasp planner: (a) no contact point found since the hammer is sunken into the table due to localization error, (b) computed contact points (red dots) are out of collision but too close to the table, (c) the planner does not allow for the contact with the table and therefore enveloping grasps are not considered resulting in an unstable grasp

controllers are crucial to ensure safety of the robot. Our approach is motivated by anecdotal experiments with human subjects which demonstrate similar behavior in human grasping of small objects (see Section 5.4). The use of compliance for grasping small objects is the main contribution of this paper.

We present three simple, yet effective, manipulation primitives for robust grasping (and releasing) of small objects from support surfaces: (1) *Compliant Finger Placement* for bringing all fingers safely in contact with the support surface, (2) *Compliant Object Grasping* for maintaining the contact between the fingertips and the support surface during the finger closure, and (3) *Compliant Object Release*. An example of landing and grasping primitives is shown in Fig.1.

2 Related Work

The literature on robotic grasping is vast. Here we only refer to the most recent related developments in the area, but we also encourage the reader to consult the extensive review in [1].

We believe uncertainty is the key challenge for grasping small objects because of limitation in perception and calibration. Many techniques attempt to handle uncertainty by explicitly modeling the contact between the fingers and the object (e.g. [13]). In practice, for small object such precise modeling is very difficult.

Deliberate interaction with the environment can be used to reduce uncertainty. Active sensing has been proposed to reduce uncertainties in object pose estimation. For

example, [7] uses pre-grasp interactions to estimate the object pose without disturbing it. Other methods such as [2, 4, 5] use deliberate interaction with the object to reduce uncertainty, e.g. in [4] a push-grasping mechanism is used to align and bring the object inside the capture envelope. These approaches apply to large objects. However, due to inaccuracies in control and perception they do not apply to small objects.

The aforementioned techniques along with other geometric grasping strategies (e.g. [1]) avoid contact with the support surface during grasp execution, and hence often fail to successfully execute a planned grasp, particularly for small objects. In contrast, we argue that reliable grasp of small object must consider contacts with support surfaces. Our approach leverages this insight. In order to allow contact with the environment compliant motion is necessary.

In concurrent related work, [8] recently presented a learning strategy to acquire manipulation and grasping skills where an initial position control policy for the manipulation task is initialized through kinesthetic demonstration. The learned policy is then augmented with a force/torque profile that is controlled in combination with position trajectories using a force compliant strategy in a closed-loop scheme that is similar to ours. The force/torque profile is learned through demonstration by optimizing a cost function which measures the task success. In contrast to their technique, our approach does not require learning for all objects at every position of the workspace, but rather implements a simple closed-loop hybrid position-force controller that generates the compliant motions necessary to maintain the proper contact between the finger tips and the supporting surface during the grasp execution, and is operational across a broad range of conditions.

We propose a compliant grasping strategy which performs pinching as well as enveloping grasps [15]. In contrast to pinching grasps, where the object is restrained by the fingertips at certain contact points, enveloping grasps are formed by wrapping the fingers and the palm around the object. The theoretical analysis (e.g. in [15] and [6]) for pulling objects from a surface into an enveloping grasp can be used when identifying grasps for new objects. The choice of the grasp is affected by various parameters including the task and the size, shape and weight of the object. Enveloping grasps are shown to be superior in terms of restraining objects as expressed in [1]. Although preferable, they often are more challenging to perform in scenarios where the object is lying down on a supporting surface, e.g., grabbing a hammer lying on a table (Fig. 1). In fact, in such scenarios, the fingers need to come in full contact with the support surface and then inserted underneath the object while the hand pushed downwards to maintain the proximity to the table. Proper control of the robotic hand to achieve such motion without breaking or stalling the fingers is the motivation for our work.

3 Hardware and Control Requirements

3.1 Hardware Requirements

The compliant motion primitives presented in this work use two sensing modalities: (1) a 3-axis force/torque sensor positioned at the wrist between the manipulator arm and the hand, and (2) strain gauges between the proximal and distal finger segments. More-

over, the proposed primitives assume knowledge of finger tip positions from forward kinematics. Force feedback from the wrist closes the loop for performing compliant motions as described below. Finger strain gauges are solely used to detect fingers contact with the support surface.

Ideally, either of the above sensing modalities can provide the necessary feedback to implement the proposed primitives. However, we found the finger strain gauges hard to calibrate and very noisy, so we only used them as a binary sensor to detect individual finger contacts. At the same time, the 3-axis force/torque was noisy enough that looking at the direction of the force vector was impractical to determine which finger had made contact, especially in the presence of kinematic modeling errors. Other sensory hardware can be used as long as the above feedback is provided. For example, tactile sensors at the tips of the fingers can be used to detect the finger contacts.

3.2 Low-Level Controllers for Hand and Fingers

3.2.1 Velocity-based Operational Space Control of Hand

We employ a velocity-based operational space formulation to generate compliant motion of the hand in response to forces seen at the wrist. This formulation is fairly straightforward to implement and demonstrates an overall good performance compared to other variations of operational space control [12]. Desired joint velocities to track a given hand velocity $\dot{\mathbf{x}}_d$, determined based on the task primitive, are calculated using Liegeois' resolved motion rate control approach as: [9]

$$\dot{\mathbf{q}}_d = \mathbf{J}^+ \dot{\mathbf{x}}_d + \lambda(\mathbf{I} - \mathbf{J}^+ \mathbf{J}) \nabla \mathbf{H}(\mathbf{q}), \quad (1)$$

where \mathbf{J} is the robot Jacobian with its pseudo-inverse denoted \mathbf{J}^+ , λ is a gain value, and $\mathbf{H}(\mathbf{q})$ is a null-space cost/utility function. Different criteria can be used to define $\mathbf{H}(\mathbf{q})$ depending on the objective, e.g., avoiding joint limits or kinematic singularities.

The desired motors torque command is calculated using the computed torque control method with an added velocity feedback [12] to track the desired joint velocities in (1),

$$\boldsymbol{\tau} = \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}}_d + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{g}(\mathbf{q}) + \mathbf{K}_{q,d}(\dot{\mathbf{q}}_d - \dot{\mathbf{q}}) \quad (2)$$

where $\mathbf{M}(\mathbf{q})$ is the inertia matrix, $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ is the Coriolis/centrifugal vector, $\mathbf{g}(\mathbf{q})$ is the gravity vector, $\mathbf{K}_{q,d}$ is a gain matrix, and $\boldsymbol{\tau}$ is the joints torque vector. The desired joint acceleration $\ddot{\mathbf{q}}_d$ is obtained by differentiating $\dot{\mathbf{q}}_d$.

In the following sections, we describe how the desired hand velocity $\dot{\mathbf{x}}_d$ is computed to achieve a compliant motion behavior in response to contact forces.

3.2.2 Position-based Control of Fingers

The fingers are controlled along their pre-defined trajectories using a position-based method. The trajectory is defined as a sequence of waypoints based on the given task primitives. This controller is used to coordinate the position of fingers along their desired trajectories during the grasping or releasing of objects.

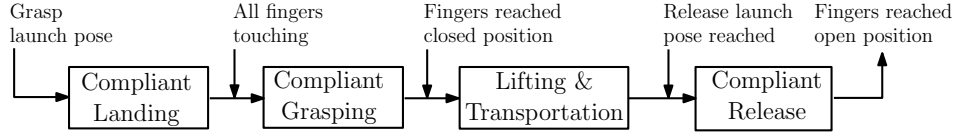


Figure 4: A grasping task decomposed into four distinct steps

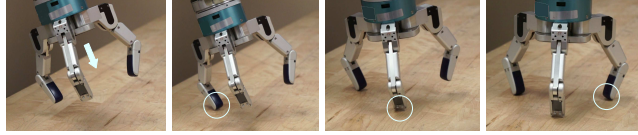


Figure 5: Compliant finger landing sequence: the circles indicate which new finger has made contact with the support surface.

4 Force Compliant Grasping Primitives

A grasping task can be decomposed into four distinct, sequentially executed steps as illustrated in Fig.4. (1) *Compliant Landing*: fingers are placed in a pre-defined grasp pre-shape and the hand is maneuvered downward until all fingers fully rest on the support surface, (2) *Compliant Grasping*: force feedback is used to maintain a desired contact force at the fingertips while the fingers joints are synchronously closed to capture the object, (3) *Lift and Transportation*: the object is lifted away from the surface and carried to the destination, (4) *Compliant Release*: the object is gently deposited on the support surface using a method similar to step 2. A similar sequence of grasping controls has been suggested in [14] inspired by neuroscience studies, but it lacks the finger landing step and the compliant primitives proposed in this paper.

4.1 Force Compliant Fingers Landing

Landing begins by positioning the palm at a safe hovering distance above the surface, in the direction of the plane’s normal. We assume that the location of the support surface and its normal are roughly known from perception.

We achieve safe finger landing by controlling the hand towards the support surface using a compliant controller. We use a velocity-based controller (Eq. 2) to generate the compliant motions. This controller brings all fingers to contact with the support surface, while preserving them from damage. Landing is achieved as soon as all fingertips establish contact with the surface. Because the fingers do not necessarily contact the surface simultaneously, we continuously update the servo control point and axis to correct the hand’s motion. This process terminates when landing is achieved (i.e., all fingers are in contact with the support surface). Figure 5 illustrates a finger landing sequence. Our landing strategy can be adapted to support various grasp pre-shapes.

Our approach relies on feedback from the finger strain gauges to determine whether a finger is in touch with the support surface. It sets contact flags C_i for fingers $i = 1, 2, 3$. These contact flags are updated in real-time during landing, and are used to

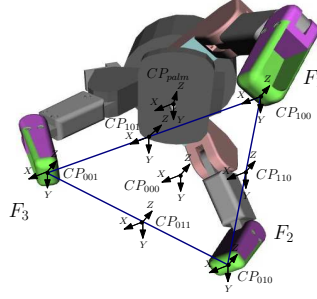


Figure 6: Control points and axes selected based on the fingertips

compute the axis around which the hand needs to be rotated. The control axes are predefined based on the selection of control points at the fingertips and their location with respect to the hand's (or the end-effector's) frame. We choose the control points at the fingertips, i.e., CP_{100} , CP_{010} or CP_{100} , as shown in Fig.6. For a single finger contact, the control axis is defined by the fingertip control point to be parallel with the support surface. When two fingers contact the surface, the mid-point between fingertips defines the control point, i.e., CP_{110} , CP_{101} or CP_{011} , and the control axis is specified by the two control points at the fingertips. The landing primitives continuously observe the contact status of the fingers and calculate the appropriate control axis around which an angular velocity ${}^{cp}\omega_d$ (with a constant magnitude) is applied to land the non-touching fingers (see Algorithm. 1).

In our experiments, uncertainties in localizing the surface normal and/or modeling errors in the robot kinematics resulted in an offset in the computation of the control axes. As a result, rotation is performed around a slightly different control axes, causing the contacting fingers to either lose contact with the surface or apply significant force onto the surface. The former leads to failure, and the latter endangers the fingers.

We rely on force compliance to avoid these risks. We move the hand in compliance with the forces exerted onto the fingers, which ensures a proper contact between the fingertips and the surface. The compliant motion introduces a linear velocity only at the control point and along the palm's normal, in response to forces measured at the wrist.

The desired force to servo the hand is calculated as

$$\mathbf{F}_d = \mathbf{F}(t) - \mathbf{F}_r - \mathbf{F}_t, \quad (3)$$

where $\mathbf{F}(t)$ is the current force seen at the wrist, \mathbf{F}_r denotes the reference force recorded *a priori* before the fingers touch the surface. The parameter \mathbf{F}_t is an attractive force value used to ensure downward motion of the hand (along its palm normal) when none of the fingers is in contact with the support surface, i.e. when $\mathbf{F}(t) \approx \mathbf{F}_r$.

Finally, the linear velocity at the control point is given as ${}^{cp}\mathbf{v}_d = \mathbf{K}_{f,p}\mathbf{F}_d$ where $\mathbf{K}_{f,p}$ is a positive gain. In practice we observed that passing \mathbf{F}_d through a deadband filter helps to decrease the oscillation effect when \mathbf{F}_d approaches zero.

The velocity screw at the control point ${}^{cp}\mathbf{x}_d$ is composed of a linear force compliant component ${}^{cp}\mathbf{v}_d$ and an angular velocity ${}^{cp}\omega_d$ around the corresponding control axis

Algorithm 1: Force Compliant Finger Landing

```
1 begin
2   Record the current finger strains,  $\mathbf{S}_{ir}$ ;
3   Set the finger contact flags to zero,  $\mathbf{C}_i \leftarrow 0$ ;
4   repeat
5     Get the current finger strains,  $\mathbf{S}_i$ ;
6     Get the current wrist forces,  $\mathbf{F}$ ;
7     Update the fingers contact flags,  $\mathbf{C}_i$  based on strains;
8     Determine the control point and axis, and the control point velocity screw
        ${}^{cp}\dot{\mathbf{x}}_d$ ;
9     Compute the desired hand velocity screw  $\dot{\mathbf{x}}_d$ ; Apply the hand velocity  $\dot{\mathbf{x}}_d$ 
       to joints using Eq.2;
10  until  $\mathbf{C}_i == 1$  for  $i = 1, 2, 3$ ;
11 end
```

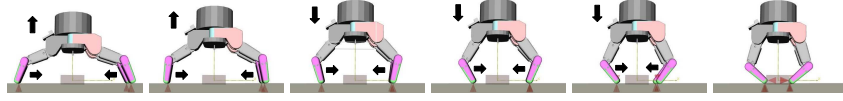


Figure 7: Force compliant grasping primitive: the hand is controlled in compliance with contact forces exerted from the support surface to the finger tips while the fingers follow their pre-defined trajectory to reach the object.

as explained above, i.e., ${}^{cp}\mathbf{x}_d = [{}^{cp}\mathbf{v}_d^T, {}^{cp}\boldsymbol{\omega}_d^T]^T$. The control point velocity screw ${}^{cp}\mathbf{x}_d$ is transformed to the corresponding hand velocity screw $\dot{\mathbf{x}}_d$ given kinematics of the fingers and the current position of control point. Finally, the hand velocity screw is then applied using the computed torque method as (2). Algorithm 1 summarizes the proposed landing strategy.

4.2 Force Compliant Grasping

Our compliant grasping strategy begins with the assumption that all fingertips are in contact with the support surface. This strategy consists of two closed-loop controllers that run independently and in parallel (see Algorithm 2). The first serves the hand using a velocity-based operation space controller. The second is a position-based controller that moves the fingers along their pre-defined trajectories (see Section 3.2).

The velocity-based operational space controller is similar to 2, which is described in the previous section. This controller generates the compliant motion of the hand in response to forces exerted by the support surface to the fingertips. The goal of this controller is to maintain contact between the fingers tips and the support surface, while closing the fingers towards the desired object-caging configuration (see Fig. 7)

The compliant motion applied to the hand is composed of only a linear motion

Algorithm 2: Force Compliant Grasping

```
1 begin
2   Record the current wrist forces,  $\mathbf{F}_r$ ;
3   Start fingers trajectory tracking by sending the first waypoints to fingers
   controller;
4   repeat
5     Get the current wrist forces,  $\mathbf{F}$ ;
6     Compute the desired hand velocity  $\dot{\mathbf{x}}_d$  using Eq.4;
7     Apply the hand velocity  $\dot{\mathbf{x}}_d$  to joints using Eq.2;
8     if all fingers reached their waypoints then
9       Send the next fingers waypoint positions;
10    end
11  until all fingers reached their desired final positions;
12 end
```

calculated at the hand frame as

$$\dot{\mathbf{v}}_d = \mathbf{K}_{f,p} \mathbf{F}_d \quad (4)$$

where \mathbf{F}_d is defined in (3). The hand velocity screw is then given as $\dot{\mathbf{x}}_d = [\mathbf{v}_d^T, \mathbf{0}^T]^T$ which is applied to the joints using the computed torque in (2).

While the fingers move along their predefined trajectories, our compliant velocity controller responds to forces that are due to the fingers' contact with the support surface. These forces are measured at the wrist. The fingers' positions are coordinated to ensure proper caging of the object, without missing it. To coordinate the fingers' positions, as soon as all fingers reach their waypoints, new waypoints are provided to the position-based controller (our second controller). Due to uneven contacts, some fingers may be lagging behind. This coordination strategy is key for successful grasping of small objects.

Our compliant grasping strategy can be used for performing both pinching and enveloping grasps. In a pinching grasp, the object is restrained by the fingertips only. We achieve this behavior by stopping the fingers as soon as contact with the object is detected via strain feedback. The enveloping grasp continues after pinching the object. It applies additional torque to the fingers while pushing hand downward, which in practice encourages the object to slip towards the palm. At the same time, we close the fingers to fully capture the object.

4.3 Force Compliant Object Release

The ability to accurately place an object and release it from grasp is as important as the ability of grasp and lift it, particularly for tasks such as stacking objects, assembly, or exchanging objects between hands in bi-manual manipulation. We devise a compliant strategy for gentle release and placement of a grasped object on a support surface. The proposed approach effectively avoids abrupt release of object from grasp and ensures gentle placement on the support surface from both precision and envelop-

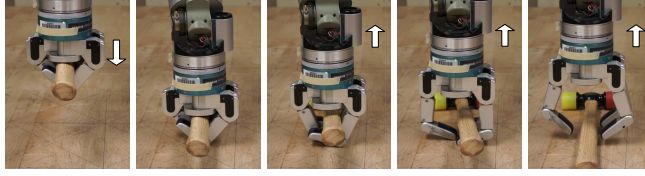


Figure 8: Force compliant release/placement primitive

ing grasps. Release from enveloping grasps is more challenging due to the inevitable, extensive contact between the fingers and support surface during the release. The proposed release/placement primitive is inspired by human release skills and utilizes the same methodology we employed to develop the compliant grasping primitive in above. The main idea is to servo control the hand in compliant to forces exerted to fingers as they open to release the object (see Fig.8).

Assuming the hand (with the object grasped) is located above the support surface, the release primitive begins by servoing the hand downward until contact with the surface is detected via continuous thresholding of the force seen at the wrist. Relying on the contact as a signal to stop the hand motion is inspired by human release strategy and has been used previously to trigger releasing objects (e.g., [14]).

Once contact between the hand/fingers and the support surface is established, we proceed by opening the fingers while compliantly servo-controlling the hand from the support surface. This is achieved by running two concurrent controllers: (1) a velocity-based operational space controller to control the hand, and (2) a position-based controller to open the fingers along their pre-defined trajectories. The proposed strategy follows the same scheme presented in Algorithm 2 with a minor difference in the desired force \mathbf{F}_d which is calculated as $\mathbf{F}_d = \mathbf{F}(t) - \mathbf{F}_r$, where the reference force \mathbf{F}_r is recorded at the time of the initial contact between the hand/fingers and the support surface.

5 Experiments and Results

To validate and demonstrate the robustness and effectiveness of the proposed grasping primitives we have performed extensive experiments on a fully integrated robotic manipulation system: a compliant 7-DOF Barrett Whole-Arm Manipulator (WAM) equipped with a Barrett 3-finger dexterous hand BH-280, and an integrated perception system. The positioning accuracy of the perception system varies depending on the object of interest. Overall, our system is capable of providing object pose with an accuracy of about 1 – 2 centimeters and 5 – 10 degrees error in position and orientation, respectively. In Section 3.1 we described the minimum hardware/Control requirements.

All objects in our experiments were located in rest poses on a table top localized by the perception system. For each of the objects in our grasping experiments the vision system provides for an object a “launch” pose for the hand where the grasp is to be initiated. The launch pose is calculated in a way to restrict the plane specified by the

fingertips parallel to the support surface with the hand centered above the object (or the target location for the release of object).

To grasp the object the hand assumes a predefined pre-shape weakly dependent on object geometry. Based on our empirical observations and given the flexibility of our proposed grasping approach, one could choose the same grasp pre-shape for objects with similar geometries. For example, for all objects with cylindrical shape (e.g., pen, screwdriver, hammer, pipe, etc.) we used a cup-like grasp pre-shape. In fact, we have observed that the cup-like pre-shape works remarkably well for grasping many of small objects we used in our experiments. This also highly conforms with our anecdotal human behavior observations. The width of fingers opening to form the cup pre-shape can be chosen arbitrarily wide as long as it satisfies the localization accuracy of the perception system to make sure that the object can be caged.

We present the experimental results and our empirical observations in three categories: fingers placement/landing, object grasping, and object release/placement experiments as follows. A video of the grasp sequence is available online at <http://youtu.be/gxaXCYY87Z0>.

5.1 Fingers Landing Experiments

An example of finger landing/placement experiment for grasping a hammer from a table top is shown in Fig.9. Initially the hand is at its launch pose centered above the hammer, see Fig.9(a). As it is seen the fingertips plane (or the hand palm) is not parallel to the table due to misalignment of the hand caused by uncertainties combined in perception and the robot kinematics. Clearly approaching the hammer along this orientation will not place all fingers in contact with the support surface (see Fig.9(b)), and hence will not yield a stable and robust grasp if executed. To fix the hand orientation and ensure all fingers are in touch with support surface, the proposed fingers landing approach servo control the hand around appropriate control axes (as described in Section 4.1) based on the current fingers touch states determined from the fingers strain feedbacks. For example, in this experiment, the hand is initially servo controlled along the normal to fingertips plane (Fig.9(a)) until the touch between finger F_2 and support surface is detected (Fig.9(b)). The strain threshold to identify the touch is 50 as indicated in the plot. Next the hand is servo controlled around the control axis at finger F_2 (Fig.9(c)) until finger F_3 reaches the support surface. Finally, the hand rotates around the control axis between fingers F_2 and F_3 until finger F_1 contacts the surface.

The plots in Fig.9 show the fingers current contact status and the fingers which are expected to contact the table next at each instant of time. The fingers strain values along with the hand linear (along the normal of fingertips plane) and angular velocities during the whole landing process are also shown. Due to kinematic errors in calculating the control axis extra force might be exerted to a finger as the hand rotates to land other fingers. For example, the strain in finger F_2 increases even after its first contact with the support surface as shown in Fig. 9. However, the force compliant motion incorporated in the proposed landing technique prevents excessive force to be applied the fingers by moving the hand away from the surface to decrease the strain on fingers and avoid damaging them.

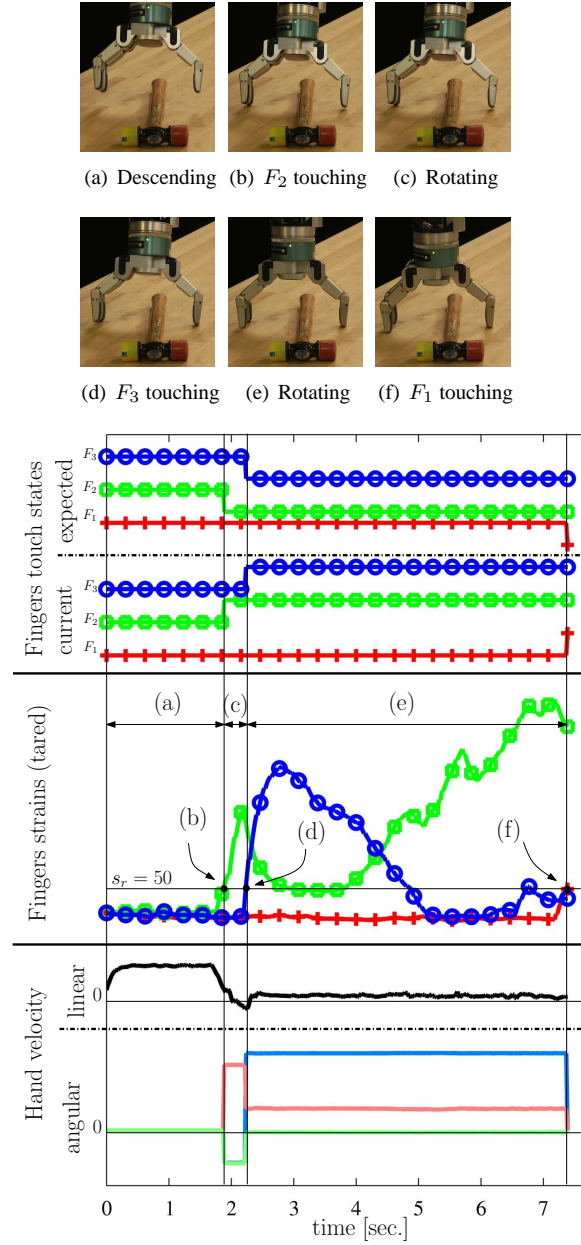


Figure 9: Compliant fingers landing

The proposed landing primitive can be used for landing fingers from different pre-shapes, e.g., Fig.10 shows landing experiments from various grasp pre-shapes. For every grasping experiment the fingers landing is first executed to ensure contact between

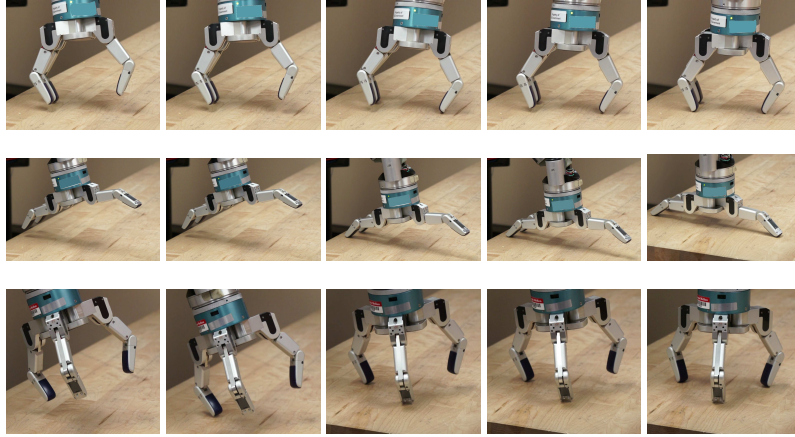


Figure 10: Compliant fingers landing/placement experiments from an open pre-shape

all the fingers and the supporting surface. This is a key prerequisite to the robustness and success of our proposed compliant grasping approach as shown in the experiments which follow. This is also a unique and novel strategy which may find applications in other robotic manipulation tasks to place fingers on the surface of objects in a compliant scheme.

5.2 Object Grasping Experiments

Each grasping experiment starts off with all the fingers initially in contact with the supporting surface. This is ensured using the fingers landing strategy presented above. Figure 11 shows snapshots of a grasping experiment to grab a pipe from a table top. The plot shows a number of parameters including: status of fingers being stuck or reached their waypoints, fingers trajectories, computed compliant linear velocity of the hand along its palm normal, and the tared force applied to the hand along its palm normal. As shown the hand responds compliantly to the forces as the fingers close along their trajectories. For example, between seconds 6 to 8 all the fingers have reached the body of the pipe (Fig.11(c)) and apply forces to the pipe body as well as the table to manage to go underneath the pipe which happens to around second 8. At this moment the forces applied by the fingers pushes the hand upward while lifting the pipe and managing to fully cage it. The compliant motion of the hand plays a crucial role to prevent damages to the fingers. For example, there are other instances (about seconds 2 and 4) where the fingers are stuck but the compliant motion does not let the manipulator to exert excessive pressure to the finger tips. Many of the state of the art grasping techniques either try to avoid the contact between the robot and support surface or are not capable of accounting for such contact, and hence fully ignore it.

The experiments presented here show that the coordination between the fingers and the hand movement plays a crucial role in achieving robust grasp of small objects. The simple yet effective compliant grasping primitive presented in this work successfully

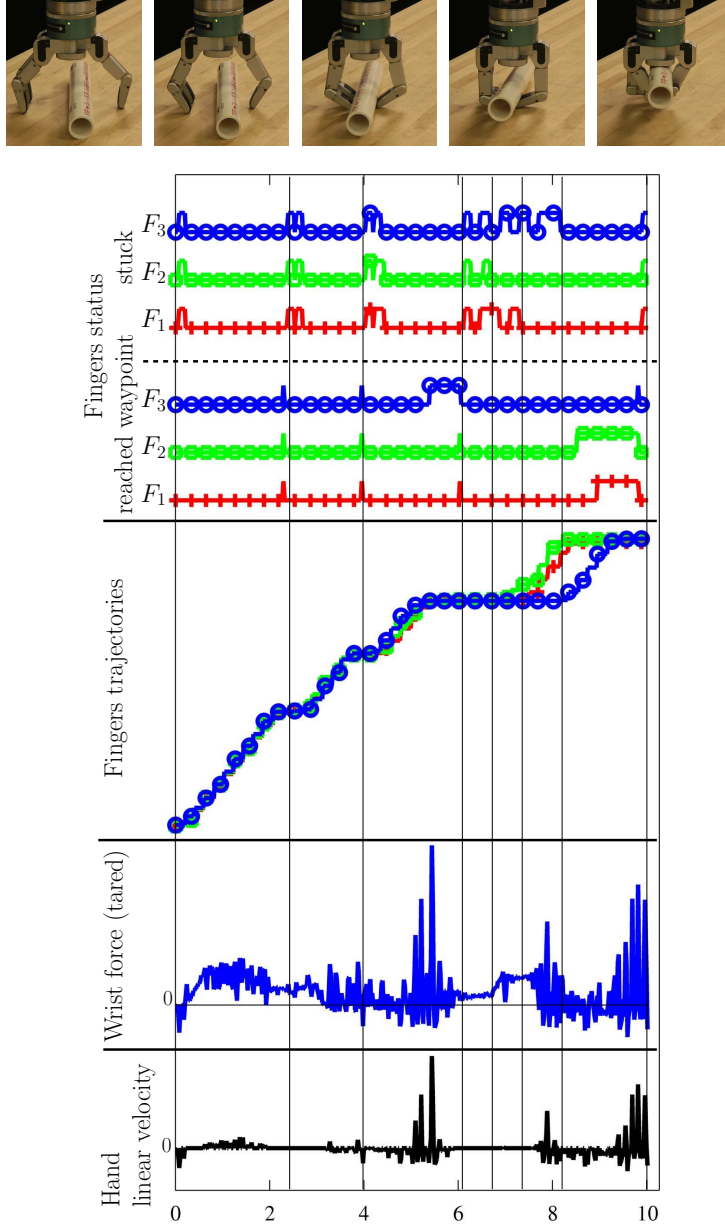


Figure 11: Force compliant grasping of a pipe from a table top

achieve this goal as shown through numerous experiments. Figure 12 shows a representative of experiments we have performed to grasp a variety of small objects, a screwdriver, a pen, and a cellphone, etc. As shown the grasp used is a precision grasp

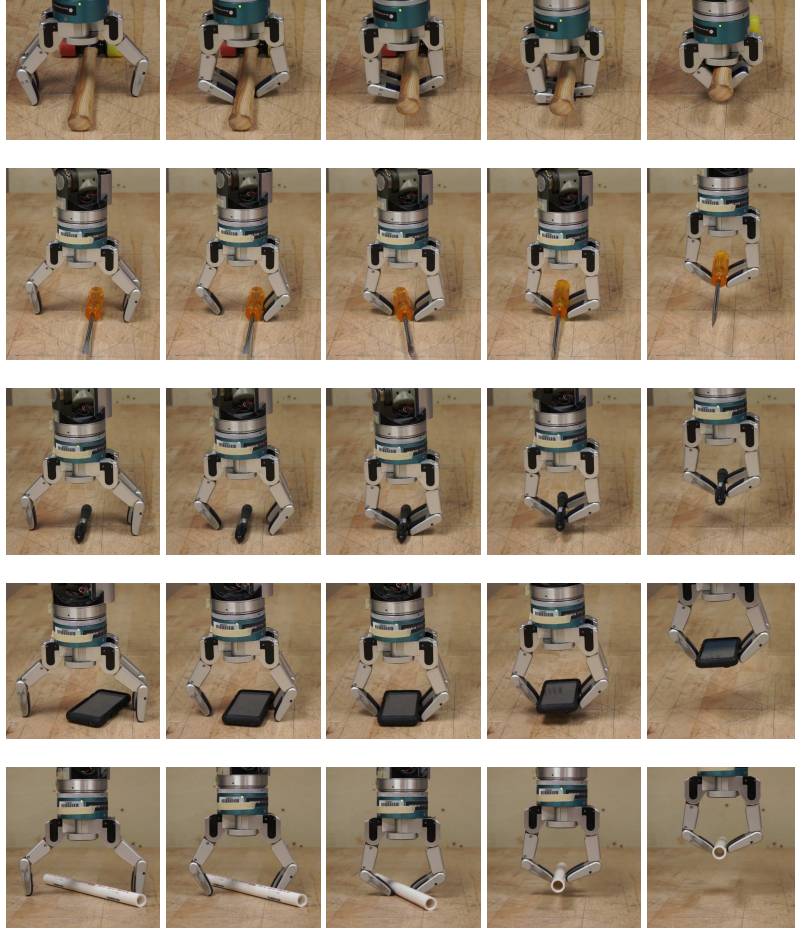


Figure 12: Examples of force compliant grasping experiments

using the fingertips to restrain the object. This was not achievable without maintaining contact between the fingertips and the support surface and coordination with the hand movement; the behaviors which are fully integrated into our compliant grasping approach. Moreover, to compensate for the uncertainties in posing the fingertips with the support surface the fingers landing primitive was used a priori to the grasp execution.

5.3 Object Release Experiments

The last set of our experiments demonstrate the compliant object release and placement. The proposed release strategy has been successfully applied to release objects from both enveloping as well as precision (fingertip) grasps. Due to space limitation we do not provide the plots explaining the details of the release strategy. However, as we noted before the release strategy heavily borrows from the compliant grasping

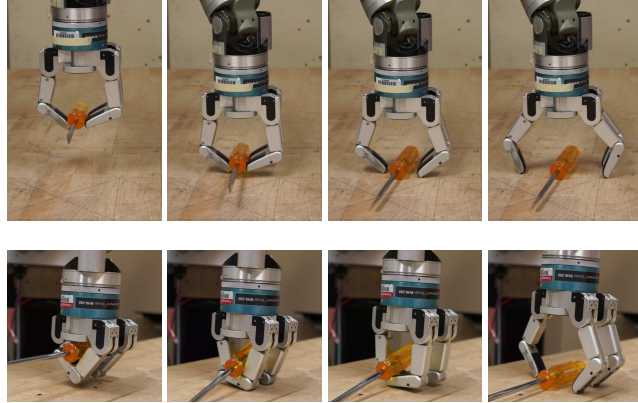


Figure 13: Examples of compliant object release/placement experiments

Statistics	Maglite	Screwdriver	Hammer
Num. Trials	24	30	35
Success Rate	92%	93%	97%

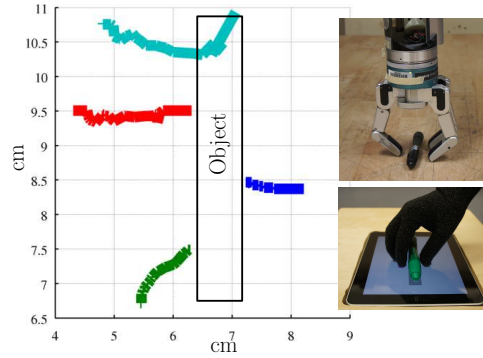
Table 1: Compliant grasp performance on sample objects

technique and one can view it as a grasp execution but in reverse order. Two representative examples of the experiments from numerous experiments which we performed are shown in Fig.13. The proposed release technique is highly robust to the uncertainties in localizing the height and orientation of the support surface and manages to gently release and replace objects on the support surface without damaging the fingers.

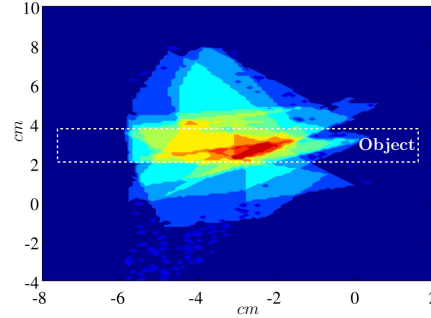
We conducted a series of grasp repeatability experiments on common objects found in our lab; namely a D-battery Maglite, a foot-long screwdriver, and a standard hammer. The objects were randomly positioned on a table, within the robot’s reach. The perception system was first used to localize the object, followed by the sequence of motions to position the hand at the grasp launch pose. From that point, the sequence of *landing*, *grasping*, *transportation* and *release* was executed, and success rate measured. The results are shown in Table1. With successes of 92% and above proves robust. Note that the vast majority of the failures were due to early testing failures, where faults in our finger reset procedures caused fingers to jam when contacting the table. Other independent, and competitive experiments, were conducted using a different robot, but using software we provided. For those tests, careful attention to finger calibration was given and our success rate for the objects mentioned was 100%, even on particular objects not previously encountered.

5.4 Parallel with Human Grasping

Complete results from controlled experiments aimed at comparing our proposed robotics grasping approach to the way humans grasp small objects is not the focus of this paper. Nonetheless, our inspiration came from careful recording of the trajectories of human grasps of various objects. The intent of our anecdotal experiments was to validate our belief that people make extensive use of contacts with the environment to reduce hand position uncertainty and move fingers into positions during grasps. We present results from a typical human experiment trial in Fig.13, in which a human subject was asked to grasp a marker. We note that in every one of over 50 trials, humans subjects make extensive contact with a supporting surface while grasping such objects, in contrast with the classical grasping approach of finger placement without other object contact¹.



(a) Left: fingertips trajectories, right: examples of robot and human grasps



(b) 2-D histogram of fingertips trajectories

Figure 14: Experiments studying human grasping strategies

Figure 14(b) shows the 2-D histogram of fingertips trajectories for 50 grasps performed by human subjects. The center of the figure shows the highest density of con-

¹A custom iPad application in an embedded test platform, with all test subjects wearing *Touchscreen Gloves*, was used to carefully measure both location and duration of contacts while grasping objects of interest.

tacts due to subjects dragging their fingers towards the center of the object, with some trial starting up to 4 cm away from the object.

For all the test subjects, we noticed that they were actually touching the surface over 2 – 4 centimeters prior to grasping the pen. Those early results appear to validate our intuition for developing the simple touch-based primitives in this work.

6 Conclusion

We addressed the problem of grasping objects that are small relative to an anthropomorphic hand, including a pen, screwdriver, cellphone, and hammer from their natural poses on a supporting surface, e.g., a table top. We argue that contact with support surfaces is critical for grasping small objects. We devised three simple, yet effective, manipulation primitives for robust grasping (and releasing) of small objects from support surfaces: (1) *Compliant Finger Placement* for bringing all fingers safely in contact with the support surface, (2) *Compliant Object Grasping* for maintaining the contact between the fingertips and the support surface during the finger closure, and (3) *Compliant Object Release*.

We conducted extensive grasping experiments on a variety of small objects with similar shape and size. The results demonstrates that our approach is robust to localization uncertainties and highlights the benefits of compliant, contact driven control strategies for grasping tasks.

References

- [1] A. Bicchi and V. Kumar. Robotic grasping and contact: a review. In *ICRA*, pages 348–353, 2000.
- [2] R.C. Brost. Automatic grasp planning in the presence of uncertainty. *Int. J. Robot. Res.*, 7(1):3 – 17, 1988.
- [3] Rosen Diankov. *Automated Construction of Robotic Manipulation Programs*. PhD thesis, Carnegie Mellon University, Robotics Institute, August 2010.
- [4] M.R. Dogar and S.S. Srinivasa. Push-grasping with dexterous hands: Mechanics and a method. In *IROS*, pages 2123 –2130, 2010.
- [5] M.A. Erdmann and M.T. Mason. An exploration of sensorless manipulation. *IEEE J. Robot. Autom.*, 4(4):369 – 79, 1988.
- [6] K. Harada, M. Kaneko, and T. Tsujii. Rolling-based manipulation for multiple objects. *IEEE Trans. Robot. Autom.*, 16(5):457 – 68, 2000.
- [7] K. Hsiao, L. Kaelbling, and T. Lozano-Perez. Robust grasping under object pose uncertainty. *Autonomous Robots*, 31(2-3):1–16, 2011.
- [8] Mrinal Kalakrishnan, Ludovic Righetti, Peter Pastor, and Stefan Schaal. Learning force control policies for compliant manipulation. In *IROS*, pages 4639–4644, 2011.
- [9] A. Liegeois. Automatic supervisory control of the configuration and behavior of multibody mechanisms. *IEEE Trans. Syst. Man Cybern.*, SMC-7(12):868 – 71, 1977.
- [10] Matthew T. Mason. Compliance and force control for computer controlled manipulators. *IEEE Trans. Syst. Man Cybern.*, 11(6):418 –432, 1981.
- [11] A.T. Miller and P.K. Allen. Graspit! a versatile simulator for robotic grasping. *IEEE Robot. Autom. Mag.*, 11(4):110 – 22, 2004.
- [12] J. Nakanishi, R. Cory, M. Mistry, J. Peters, and S. Schaal. Operational space control: a theoretical and empirical comparison. *Int. J. Robot. Res.*, 27(6):737 – 57, 2008.
- [13] Van-Duc Nguyen. Constructing stable grasps. *Int. J. Robot. Res.*, 8(1):26 – 37, 1989.
- [14] J.M. Romano, K. Hsiao, G. Niemeyer, S. Chitta, and K.J. Kuchenbecker. Human-inspired robotic grasp control with tactile sensing. *IEEE Trans. Robot.*, 27(6):1067 –1079, 2011.
- [15] J.C. Trinkle, J. M. Abel, and R. P. Paul. An investigation of frictionless enveloping grasping in the plane. *Int. J. Robot. Res.*, 7(3):33–51, June 1988.