UNIVERSITÀ
DEGLI STUDI
DI GENOVA

ISTITUTO ITALIANO
DI TECNOLOGIA

UNIVERSITÀ DEGLI STUDI DI GENOVA

DOCTORAL THESIS

Robotic Manipulation using Tactile Feedback

Author:
Massimo REGOLI

Supervisors:
Dr. Lorenzo NATALE
Prof. Giorgio METTA

A thesis submitted in fulfillment of the requirements
for the degree of Doctor of Philosophy

in the

Humanoid Sensing and Perception - iCub Facility
Istituto Italiano di Tecnologia

Declaration of Authorship

I, Massimo REGOLI, declare that this thesis titled, "Robotic Manipulation using Tactile Feedback" and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed:

Date:

Abstract

Massimo REGOLI

Robotic Manipulation using Tactile Feedback

Object manipulation is one of the main research areas in the field of robotics. Currently available robotic hand-arm systems move faster and more accurately than humans, and their sensors capture more information and at a higher precision. Even the smallest forces can be detected. Despite this, they manipulation capabilities are very poor when compared to humans. Neuroscience provides a clear reason for the superiority of human hands: during manipulation, humans make substantial use of tactile sensing.

For this reason, over the last few decades interest in dexterous robot manipulation using tactile sensors has been growing, leading to the development of several types of tactile skins based on different technologies. Tactile sensors allow to reliably estimate forces at the points of contact and detect object properties such as softness, texture and shape. These capabilities turn to be extremely useful to profitably face manipulation-based tasks such as the ones treated in this thesis, namely grasp stabilization and in-hand object recognition.

Stable object grasping is a complex task and poses some very intricate problems. One of the main difficulties is to find grasping strategies and algorithms which generalize to new situations and thus are sufficiently robust against uncertainties. With this respect, in this work we present a grasp controller which combine techniques from machine learning and control theory to successfully perform a stable grasp on unknown objects of different shape, size and material. Differently from other works, our method allows for independent control of the grip strength, which is beneficial for several tasks, like slip control or object softness exploration.

Like stable object grasping, also in-hand object recognition is a very popular topic in robotics. While vision was originally the main source of information, now researchers agree that exploiting tactile sensors is crucial to

successfully achieve this task. Indeed, vision is strongly affected by lighting conditions, while object properties such as softness can only be detect by applying active tactile exploration. In this thesis we describe a novel method for in-hand object recognition which exploits the stable grasp controller to reposition the object in a repeatable way and perform two exploratory behaviours, namely, squeezing the object to get information about its softness and wrapping all the fingers around the object to get information about its shape. The high repeatability of the object position leads to have low variance in the collected features, which, in turn, improves the recognition accuracy of the classifier.

All the experiments performed to prove the effectiveness of the above mentioned methods were carried out on the iCub humanoid robot.

Contents

List of Figures

List of Tables

List of Abbreviations

GMM	**G**aussian **M**isture **M**odel
GURLS	**G**rand **U**nified **R**egularized **L**east **S**quares
YCB	**Y**ale **C**MU **B**erkeley
DOF	**D**egree **O**f **F**reedom
PCB	**P**rinted **C**ircuit **B**oard

Chapter 1

INTRODUCTION

1.1 Tactile sensing

The sense of touch is essential for humans. We use it constantly to interact with our environment. In particular, it makes us extremely dexterous in manipulation. People can grasp a wide variety of objects, perform complex tasks, and switch between grasps in response to new task requirements. This is allowed in part by the physical structure of our hands (multiple fingers with many degrees of freedom), and in part by our sophisticated control skills. In large measure this control capability is based on tactile and force sensing, which allows to capture properties at the finger-object contact. Indeed, people lose most of such capabilities when deprived of reliable tactile information through numbness of anesthetized or cold fingers [4].

To detect environmental properties we rely on different sensory modalities. Both vision and touch can provide information about an object's shape, but the sense of touch usually provides much more knowledge about material properties such as texture, temperature, and weight. A baby starts exploring himself and his environment through his sense of touch [5] [6], using his hands and mouth as the principal exploratory tools. Such exploration has been proved to start already in the womb [7].

As for humans, it is fundamental that robots are also endowed with advanced touch sensing in order to be aware of their surroundings and provide information for subsequent tasks such as object recognition and in-hand object manipulation, especially when vision is limited or obstructed. Originally, vision was the primary sensory modality on which robots relied to define their behavior, while interaction forces were mainly controlled by using force and torque sensors either at the joint or at the end-effector level. However, such sensors only provide an indirect measure of collisions that strongly rely on an accurate model of the robot. Moreover, contact can be detected only in

certain parts of the robot body, which implies that interaction is expected at a specific robot location, like the end-effector tip.

For more complex forms of interactions, in which the typology of contact cannot be accurately predicted or modeled in advance, force and torque sensors might not be enough. Safety, for example, would benefit of a distributed tactile "skin" able to cover the whole robot body so that contact can be detected anywhere. Such tactile systems would also be helpful for human-robot interaction. Indeed, as a human, a humanoid robot would be expected to rely on physical contact for communication (like tapping on someone's shoulder to attract his attention). Furthermore, a tactile system able to estimate location-specific contact forces and material properties would be crucial to achieve more complex tasks such as object manipulation and object recognition.

As a result, in recent decades a huge number of tactile sensors have been developed and integrated in robot skins, involving many different technologies [8]. Despite this, the topics of tactile object manipulation and recognition have not been sufficiently explored yet. The most important reason lies in the intrinsic complexity of the problem at hand. First, the great variety of tactile platforms makes it hard for researchers to find a common approach to the problem. Moreover, the signals generated by tactile sensors are generally high-dimensional and noisy, and they often require movement to carry useful information (i.e. they may require active exploration).

A strategy used to represent such amount of data is to interpret the tactile feedback as a simple "image", in which each tactile element plays the role of a pixel [9] [10]. Unfortunately, this analogy is in practice very weak. Indeed, images captured by physically different cameras are in principle equivalent, since specific geometric and optical calibration can be performed by *state-of-the-art* software. Conversely, tactile images are not based on any standard model and are strongly hardware dependent. Indeed, on each robot tactile sensors are distributed to fit its specific shape, even changing in density to provide higher resolution in body areas that requires to be more sensitive. Moreover, according to the compliance of the skin on which they are integrated, the relative position of taxels can even change as the robot moves. As a result, the way the tactile data is processed may vary considerably according to the specific platform used. In addition, tactile information can be interpreted according to the desired function of the robot. Relevant information that can be extracted from sensing data include shape, material

properties, object pose and estimated force at the contact location. In particular, such information is crucial to successfully achieve the main tasks treated in this work, that is stable object grasping and tactile object recognition.

Grasp stability is a basic requirement of most kinds of manipulation and represents a challenging problem in robotics. It needs to be robust to external perturbations and adapt to unknown objects. While performing a stable grasp, grip strength control can be a desirable property for many applications, such as slip control or object manipulation. In this thesis we will describe in detail an approach for stable object grasping and simultaneous grip strength control using tactile feedback, which is able to deal with unknown objects of different shape, size and material. We develop a generic method that exploits the structure of an anthropomorphic hand to be simple and effective. Our approach uses techniques from classical control theory to develop a controller in charge of coordinating the fingers for achieving grasp stabilization and grip strength control.

The other topic we focused on, tactile object recognition, is also very relevant in robotics. Even when vision is available, some object properties such as material and texture can be better discriminated by applying active tactile exploration. In this thesis we will illustrate a novel method for in-hand object recognition. The method is composed of the grasp stabilization controller and two exploratory behaviours used to capture the shape and the softness of an object. Grasp stabilization plays an important role in recognizing objects. First, it prevents the object from slipping and facilitates the exploration of the object. Second, reaching a stable and repeatable position adds robustness to the learning algorithm and increases invariance with respect to the way in which the robot grasps the object.

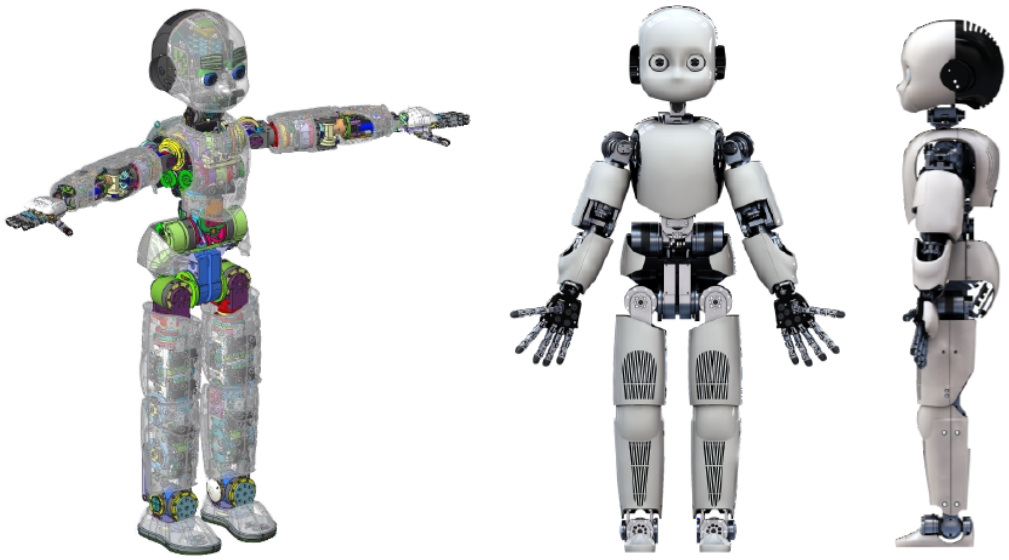
In the next sections we first list the relevant publications achieved during the Ph.D. period, then we briefly describe the platform used for our experiments, the iCub robot, focusing on its hands. Then, we dedicate chapters 2 and 3 to accurately detail our methods for, respectively, stable object grasping and tactile object recognition. In each of these chapters we provide both an introduction and a conclusion for the specific topic faced. Finally, in ch. 4 we report the general conclusions of this thesis.

1.2 Relevant publications

- **Regoli M.**, Pattacini U., Metta G. and Natale L., ***Hierarchical Grasp Controller Using Tactile Feedback***, in IEEE-RAS International Conference on Humanoid Robots, Cancun, Mexico, 2016.
- **Regoli M.**, Jamali N., Metta G. and Natale L., ***Controlled Tactile Exploration and Haptic Object Recognition***, in Proc. IEEE International Conference on Advanced Robotics, Hong Kong, China, 2017.
- Vezzani G., **Regoli M.**, Pattacini U. and Natale L., ***A Novel Pipeline for Bi-manual Handover Task***, accepted to Special issue on Advanced Manipulation, Advanced Robotics, 2017.
- Tomo T.P., **Regoli M.**, Schmitz A., Natale L., Kristanto H., Somlor S., Metta G., Sugano S., ***A New Silicone Structure for uSkin - a Soft, Distributed, Digital 3-axis Skin Sensor - and its Integration on the Humanoid Robot iCub***, in Proc. IEEE International Conference on Robotics and Automation, Brisbane, Australia, 2018.

1.3 The iCub robot

The platform used to carry out all the experiments reported in this thesis is the iCub humanoid robot [11] [12]. It is an open source cognitive robotic platform created with the purpose of promoting collaborative research in human cognition, human robot interaction, and embodied artificial intelligence. It was developed as part of the RobotCub European project, which aimed at 1) creating an open hardware/software humanoid robotic platform for research in embodied cognition and 2) advancing our understanding of natural and artificial cognitive systems by exploiting this platform in the study of the development of cognitive capabilities. Fig. 1.1 reports the iCub model. In order to better represent the characteristics of a learning child, the iCub kinematics were designed to mimic the body of a four years old child. Indeed, the robot is 104 cm tall, weights 22-25 kg (depending on the version) and has 53 DoF, most of them in the upper torso. The hands have 9 DoF each, allowing for dexterous manipulation and different grasping configurations. The head features 3 DoF in the neck, while tracking and vergence behaviors are supported by the extra 3 DoF controlling the eyes. Moreover, the torso has 3 DoF and each leg comprises 6 DoF. Joints are actuated with brushless and DC motors, controlled by embedded dedicated boards.



(A) (B)

FIGURE 1.1: The iCub humanoid robot used as platform for the experiments.

In order to properly interact with the environment, the iCub is endowed with several sensors. It has two cameras in the eyes for vision, microphones for recording sound, and a full body tactile skin system for tactile feedback. In addiction, proprioception is provided by means of an IMU located in the head, motor encoders in each of the joints, and 6 F/T sensors in shoulders, hip and ankles. The processing unit representing the "brain" of the iCub is a PC/104 form factor mounted inside the head.

1.3.1 The iCub hand

The iCub hand (see Fig. 1.2) has been specifically designed for the iCub and therefore an exceptional level of integration was achieved allowing high dexterity and sensorization in limited dimensions.

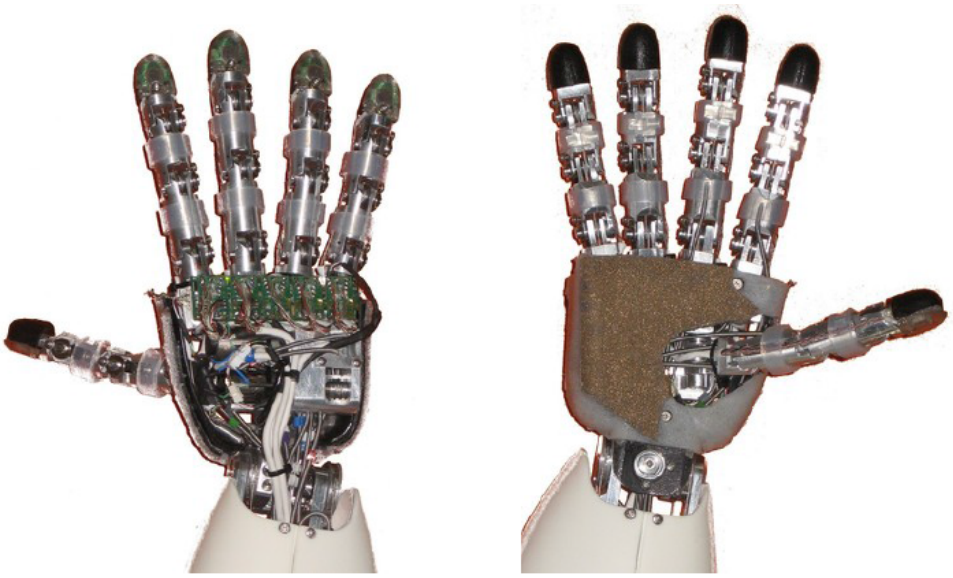


FIGURE 1.2: Two views (front and back) of the iCub hand with sensorized palm, tactile fingertip (black tips) and position sensors at all joints. Electronic for the sensors are embedded in the hand. Picture from [1].

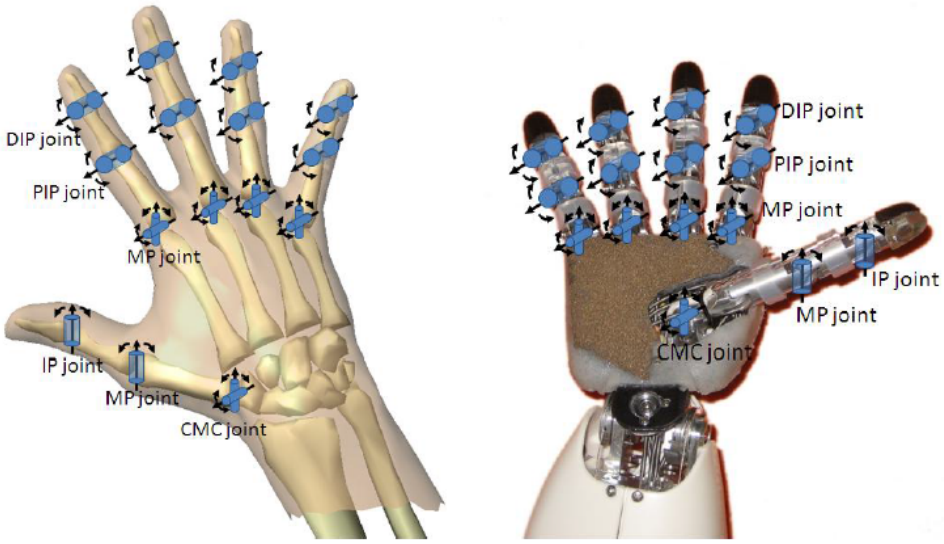


FIGURE 1.3: Left: a sketch of the human hand with all its joints; joints name for the middle, ring and little fingers are the same indicated for the index finger. Right: a picture of the iCub hand with all its joints; notice the analogy with the human hand joints. Picture from [1].

Actuation system

The hand actuation system is based on tendons. Most of the motors are integrated in the forearm and the associated tendon routed through the wrist.

Only two motors are embedded in the palm. Adopted tendon actuation can be divided into two different classes: *open-ended* tendon drives and *closed-loop* tendon drives (see [13] for details).

The hand has 20 joints organized in 9 degrees of freedom (see Fig. 1.4 and 1.5). Its dimensions (50*mm* long, 34*mm* wide at the wrist, 60*mm* wide at the fingers and 25*mm* thick) and ranges of motion were inspired by those of a human hand. In particular, excluding the palmar abduction, all the other human hand joints have been replicated. The thumb has four joints: two of them are located in the carpometacarpal (CMC) joint while the other two joints are in the metacarpophalangeal (MP) joint and in the interphalangeal (IP) joint respectively. Similarly the remaining four fingers (index, middle, ring and little) have four joints each: two located in the metacarpophalangeal (MP) joint and the other two in the proximal interphalangeal (PIP) joint and in the distal interphalangeal (DIP) joint respectively (see Fig. 1.3 for a detailed description).

The actuation of all these 20 joints is obtained using 9 DC motors (resulting in 9 DOFs) 7 of which are embedded in the forearm and 2 in the hand. Therefore, certain DOFs are obtained by coupling different joints (either tightly or elastically) so that they are moved by a single motor in a synergistic fashion. Specifically:

- A single motor embedded in the hand is used to actuate the index, ring and little fingers abduction/adduction movements.
- The thumb distal joints are actuated by an open-ended tendon drive. A torsional spring-return on both the MP and IP joints accumulates energy when the motor pulling the cable flexes the joints. This potential energy is then used when extending the joints.
- The index finger distal joints (PIP and DIP) are actuated by a single motor according to same actuation structure described for the thumb MP and IP joints. A similar structure has been adopted for the middle finger distal joints.
- The little and ring finger joints (MP flexion/extension, DIP and PIP joints) are moved by a single motor with a suitable coupling mechanism. A closed-loop tendon moves a linear slider; the slider is then used to pull simultaneously two open-ended tendon drives: one tendon moves the little finger; the other tendon actuates ring finger.

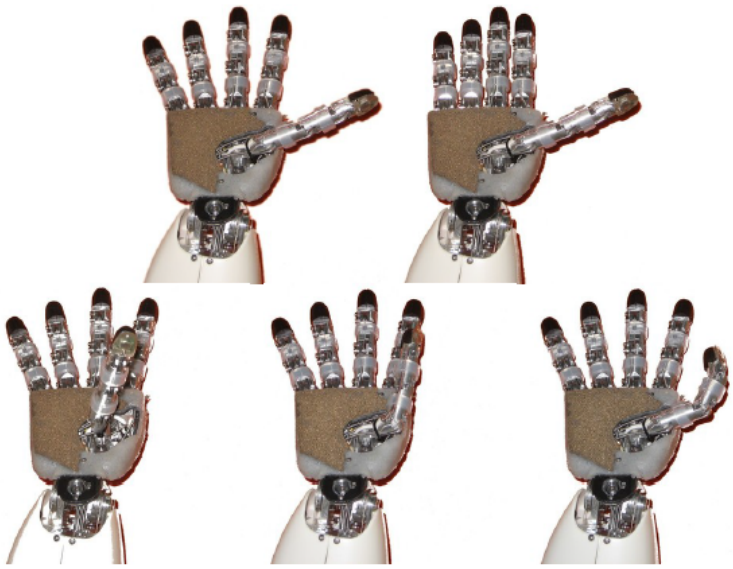


FIGURE 1.4: A sequence of images of 4 DOFs of the iCub hand. From the top-left corner (reading order): (1) initial configuration for all successive postures; (2) (index, middle, ring, little) fingers MP abduction, (3) thumb MP opposition,(4) thumb MP abduction, (5) thumb MP-IP coupling. Picture from [1].

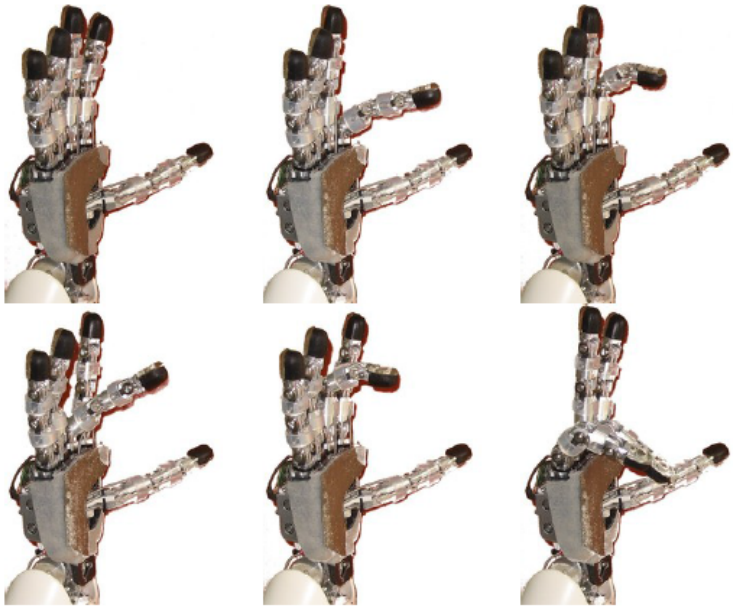


FIGURE 1.5: A sequence of images of 5 DOFs of the iCub hand. From the top-left corner (reading order): (1) initial configuration for all successive postures; (2) index MP flexion, (3) index PIP-DIP coupling, (4) middle MP flexion, (5) middle PIP-DIP coupling, (6) ring-little fingers coupling. Picture from [1].

Position sensors

The positions of the hand joints are measured with tiny Hall effect sensors (see Fig. 1.6 right) whose analog output is converted to digital by a custom made board (see Fig. 1.6 left). This board has been produced in two different (symmetric) form factors in order to be optimally embedded in the right and left hand. The 12 bits analog to digital conversion of the 17 signals relies on a 16 bit DSP Microchip and on a multiplexer. The digital data are transmitted to the control board via CAN-bus.

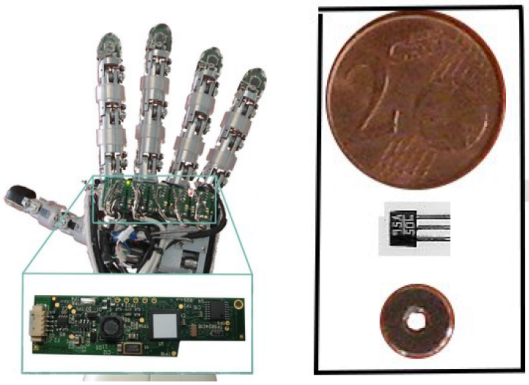


FIGURE 1.6: Left: a picture of the hand with a zoom on the embedded board designed to collect the analog data from the 17 (Hall effect) position sensors. Right: a picture of the analog Hall effect sensor together with the ring-shaped magnet used to generate the magnetic field sensed by the sensor. Magnets are positioned at each joint with the sensor underneath the magnet. Picture from [1].

Skyn sensors

We have equipped the hands of iCub with a distributed pressure sensing system based on capacitive technology. In particular, the skin of the palm incorporates four triangular modules which provide 12 pressure measurements each, and also each of the five fingertips provides 12 pressure measurements (giving the sum of 108 sensitive elements per hand. To obtain the 12 measurements, each triangular palm module and each fingertip incorporates a flexible printed circuit board (PCB), which has 12 round pads, one for each sensitive element, which are connected to a capacitive to digital converter chip, which can collect and send 12 measurements of capacitance over an I2C serial bus. The flexible PCBs of the palm can be used to cover generic curved surfaces (see Fig. 1.7), while the small size and very curved shape of the fingertips made it necessary to design a specialized PCB which can be wrapped around the inner support of the fingertip (see Fig. 1.8). Because the

PCBs provide a serial interface, the fingertip connections require only 4 wires each, which are routed along the sides of the fingers. The PCBs in the palm are also electrically connected to each other and only one of them needs to be connected with wires. All the data is collected by a small micro-controller board, which is located in the forearm of the robot and which sends the measurements via a CAN bus.

Above the flexible PCBs in the palm and the fingertips is a roughly $2mm$ thick layer of soft silicone foam (Soama Foama 15 from *Smooth-On*). It acts as a dielectric for the capacitive pressure sensor and makes the sensor compliant. On top of the foam there is a second conductive layer: electrically conductive Lycra for the palms, electrically conductive silicone for the fingertips (see Fig. 1.9). This layer is connected to ground and enables the sensor to respond to objects irrespective of their electrical properties. In addition, this layer reduces electronic noise from the environment. When pressure is applied to the sensor, this conductive layer gets closer to the round pads on the PCB and thereby changes their capacitance. We use this change in capacitance as an estimation of the pressure applied to the sensor.

The conductive silicone is an in-house made mixture of silicone CAF4 from *Rhodia-Silicones* and carbon black particles Vulcan XC72 from *Cabot*. It is sprayed on top of the silicone foam with the help of the solvent tetrahydrofuran. The conductive silicone is flexible, easily conforms to round shapes and the thin layer we apply has about $10k\Omega$ over the maximum distances found in the fingertip.

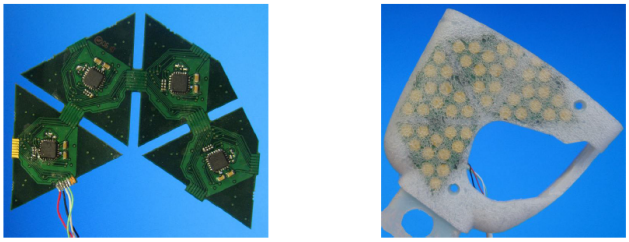


FIGURE 1.7: Left: a picture of triangular PCBs before being embedded in the palm. Each triangular module includes an AD7147 chip visible in the middle of the triangles. Right: a picture of the sensorized palm without the Lycra conductive layer. The visible white foam acts as a deformable dielectric for the capacitive pressure sensor and guarantees the compliance of the sensor. Also visible are the round pads on the triangular modules. The sensor delivers one capacitance measurements for each round pad. Picture from [1].

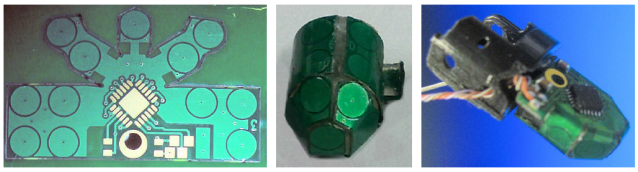


FIGURE 1.8: Left: a picture of flexible PCB before being wrapped around the fingertip. The 12 round pads for the capacitive pressure sensor system are visible as well as the soldering points for the AD7147 chip. Center: a picture of the flexible PCB wrapped around the inner support. The inner support is printed with a 3D printer (Eden 3D printer from Objet). Right: a picture of the fingertip without the fingernail and the soft silicone foam. Picture from [1].

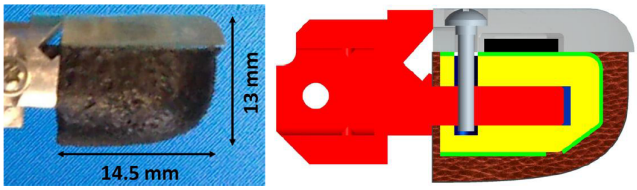


FIGURE 1.9: Left: a close-up picture of the fingertip. Right: cross-section of the fingertip. The inner support of the fingertip is shown in yellow, and the flexible PCB that is wrapped around it is depicted in green. To mechanically attach the fingertip to the hand, the last phalanx of each digit (shown in red) has a stick that fits inside a hole in the inner support. A screw is used to secure the fingertip and in addition the screw fixes a fingernail on top of the fingertip that covers the PCB. The dielectric made of silicone rubber foam is depicted in brown, and around the foam there is the carbon black layer. The AD7147 chip is also shown in black. Picture from [1].

Force estimation

The raw feedback of each taxel is read at 50 Hz and is provided in dimensionless units in the range [0,255], where greater values correspond to lower forces. However, for some of the tasks described in this thesis, such tactile data needs to be processed to compute an estimate of the global force applied by each fingertip. This is done as follows:

- At robot startup the tactile sensors calibration is performed: the tactile output is collected for 2 seconds and for every taxel the mean is computed and subtracted from any further tactile reading, so that when no pressure is applied on the skin, the output is close to zero for every taxel. The resultant value is then multiplied by -1 so that higher pressures correspond to higher values.

- In order to counteract the effect of noise in the data the output is smoothed out by applying an exponential moving average.
- Finally, we estimate the global force applied at each fingertip by taking the magnitude of the vector obtained by summing up all the normals at the 12 sensor locations weighted by the related sensors response, precessed as explained in the previous steps.

Chapter 2

STABLE GRASPING

2.1 Introduction

Grasp stability is a fundamental topic in robotics. A stable grasp is needed in order to prevent objects from slipping and is the basis for manipulation. In order to achieve this objective, many works adopt analytical solutions (see [14] for a review). Indeed, when the hand kinematics and the contacts between the hand and the object are known, it is possible to determine if the grasp is in force or form closure [15], which is sufficient for stability. However, many difficulties arise when the object model is unavailable or partially known. As a result, grasps cannot be pre-planned, and the typical strategy is to make extensive use of sensors in order to deal in real-time with environmental uncertainties [16].

In this context, tactile feedback can reveal object properties which could be hardly detected by other sensors (e.g., object softness and shape at the points of contact) and perform proper reactive strategies. Different works pointed out the importance of exploiting such a rich set of sensory information while manipulating objects [17][18]. Dang *et al.* [19] simulate tactile feedback using a soft finger contact model in *GraspIt!*, a robotic grasping simulator [20]. Using the simulation technique, they compute the tactile contacts of thousands of stable grasps, proving that the tactile feedback along with the hand kinematic data carry meaningful information for the prediction of the stability of a blind robotic grasp.

One way to approach the problem is by focusing on the selection of feasible points of contact to avoid unstable grasps. Hsiao *et al.* [21] propose a robust approach for unknown objects that makes use of visual sensors to determine a ranked set of grasps, using heuristics based on both the overall shape of the object and its local features. Then, a reactive grasping strategy exploits tactile feedback to execute a compliant robust grasp. Felip *et al.* [22] developed a grasp controller that deals robustly with uncertainty using feedback

from different contact-based sensors. This controller assumes a description of grasp consisting of a primitive that only determines the initial configuration of the hand and the control law to be used. Ciocarlie *et al.* [23] consider, as research in neuroscience has shown, that the human hand during grasping is dominated by movement in a configuration space of highly reduced dimensionality. They extend this concept to robotic hands in order to derive optimization algorithms that simplify the task of finding stable grasps even for highly complex hand designs.

However, typically these methods rely, at least partially, on vision or make assumptions about the object model. Other works still aim at evaluating the grasp stability but do not provide active strategies to improve the grasp. For example, Schill *et al.* [24] propose an approach for grasp stability learning, which estimates the stability continuously during the grasp attempt. This is opposed to usual approaches, that analyze the tactile sensor readings only at the end of the grasp attempt, which makes them somewhat time consuming.

Another approach – considered in my work – is to adjust an initial unstable grasp to a stable one. In this respect, Steffen *et al.* [25] present a tactile-driven method that dynamically uses the robot's grasping experience to adapt the grasping control by targeting the best matching posture from the experience base. To efficiently represent the experience, they introduce a grasp manifold and provide approximations using self-organizing maps. Ascari *et al.* [26] makes use of cellular neural networks to process a large amount of tactile sensing signals. They extract both hand shape and spatial-temporal features in order to control the sensory-motor coordination of the robotic system. Dang *et al.* [27] select a series of shape primitives to parametrize potential local geometries which novel objects may share in common. Then they build a tactile experience database that stores information of stable grasps on these local geometries and exploit it to guide a grasp adjustment process while grasping novel objects around similar local geometries.

In these kind of approaches many times stable grasps are learned with the help of a human demonstrator. Indeed, learning by demonstration proves to be helpful in order to reduce the complexity of tasks where many variables are involved [28][29]. Sauser *et al.* [30] introduce an approach for grasp adaptation which learns a statistical model to adapt hand posture solely based on the perceived contact between the object and fingers. Specifically, they infer the hand configuration and pressure at the fingertips given the estimated normals of contact between the hand and the object. Li *et al.* [31] use a similar model to develop a grasp stability estimator based on an object-level

impedance controller. It is used to regulate the stiffness at the fingertips depending on the tactile readings and on the relative positions of the points of contact. Huang *et al.* [32] generate a set of stable grasping demonstrations for a given object and a robot hand, build a Gaussian Mixture Model for the training dataset offline, and then use the model to quickly generate a new grasp given a starting object-hand configuration.

All these grasp stability approaches implicitly define the grip strength to be applied to the object. This is sufficient if the only objective is to achieve a stable grasp, however, it strongly limits any further grip strength control on the object. An independent control of the grip strength is beneficial for several tasks, like slip control or object exploration (e.g., to explore object properties like softness and type of material). However, the problem of controlling the grip strength while maintaining a stable grasp is hard. Indeed, due to several nonlinearities in the system, a simple proportional variation of the forces applied to the object does not guarantee that stability is maintained. Gunji *et al.* [33] propose a method for controlling grasping force to resist tangential force applied to the grasped object using a feedback control system with the tactile sensors output. Despite the grip strength is actually controlled in order to avoid slip, it cannot be arbitrarily chosen. Jalani *et al.* [34] achieve grip strength control using a model reference approach where a virtual mass-spring damper system is used to design a robust active compliant control. However, the model parameters need to be tuned for every different object class.

In this work we combine techniques from control theory and machine learning in a hierarchical control. The novelty of our method is that it decouples the problem of grip strength control and grasp stability, providing an effective framework where both objectives are achieved at the same time. Our solution can be applied to unknown objects of different size, shape and material, without the need for object specific tuning. We deal with precision grasps [35], where only the fingertips are in contact with the object.

We validated our method on the humanoid robot iCub [11], performing experiments to demonstrate reliable control of grip strength and improvement of grasp stability.

In the next section we present the methodology used to solve the problem. In section 2.3 we describe the platform, the experiments carried out to validate our method and the related results. Then, in section 2.4 we show how the method has been applied and extended to achieve even more complex tasks. Finally, in section 2.5 we draw the conclusions of this work.

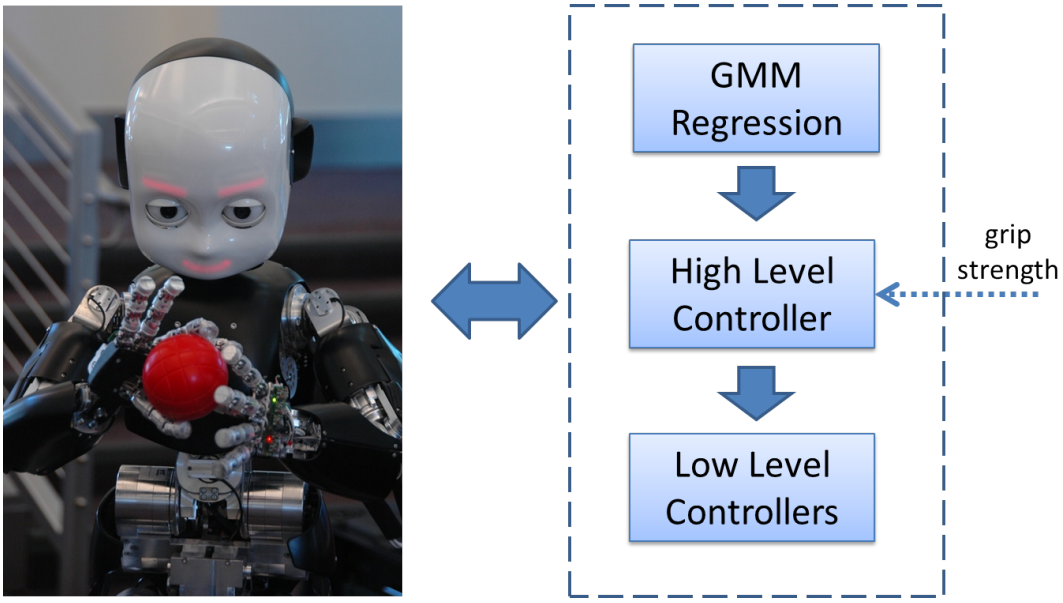


FIGURE 2.1: Schema of our hierarchical approach. Arrows represent information flow.

2.2 Methodology

We propose a hierarchical method made of three main components (Fig. 2.1):

- A low-level controller framework, composed of a force controller for each finger.
- A high-level controller, which determines the force reference values for each finger in order to stabilize the grasp while maintaining a given grip strength. At this stage, the controller is simplified by taking advantage of the anthropomorphic structure of the hand.
- A Gaussian mixture regression model, which exploits the high-level controller to further improve stability by changing the hand configuration. We made use of learning by demonstration in order to describe the space of stable grasps.

The data required for the Gaussian mixture model (GMM) training process are significantly reduced in quantity with respect to other methods considering that the relationship between the forces at the fingertips do not need to be learned, since the underlying high-level controller takes care of that.

In this work we only focus on three-finger precision grasps. We make extensive use of tactile feedback, and we define the vector $\mathbf{f} \equiv [f_{th}\ f_{ind}\ f_{mid}] \in \mathbb{R}^3$ containing the tactile reading at each fingertip. Since we deal with precision grasps, the palm is not taken into consideration.

In the following subsections we first give our definition of grip strength for anthropomorphic hands, then we detail our strategy.

2.2.1 Grip strength

Inspired by work on humans [36], we define the grip strength as the measure of force exerted on the object by the thumb (on one side) and the index and middle fingers (on the other side). Ideally, the tactile readings f_i at each fingertip are proportional to the magnitude, F_i, of the real forces. However, our tactile sensors are subject to calibration errors, noise, hysteresis, and they may detect only normal forces. Therefore, we model f_i as:

$$f_i = k \cdot F_i + e, \tag{2.1}$$

where k is a proportional value converting the tactile sensors output (in our case capacitance) into forces (newtons), while e represents a random variable having a normal distribution and equal variance σ on each fingertip, i.e., $e \backsim \mathcal{N}(0, \sigma)$, $f_i \backsim \mathcal{N}(k \cdot F_i, \sigma)$. In addition, we define f_{IM} as the sum of the tactile readings at the index and middle fingertips and F_{IM} as its related real force magnitude, i.e. $f_{IM} = f_{ind} + f_{mid} \backsim \mathcal{N}(k \cdot F_{IM}, 2\sigma)$.

We further assume that when the grasp is stable, the directions of all applied forces are parallel (Fig. 2.2) and, as a result, F_{th} must be equal to F_{IM}. Under this assumption we define the tactile measure of the actual grip strength as $g = k \cdot F_{th} = k \cdot F_{IM}$, and its estimate $\hat{g}(\mathbf{f})$ as the most probable value of g given the observations f_{th} and f_{IM}:

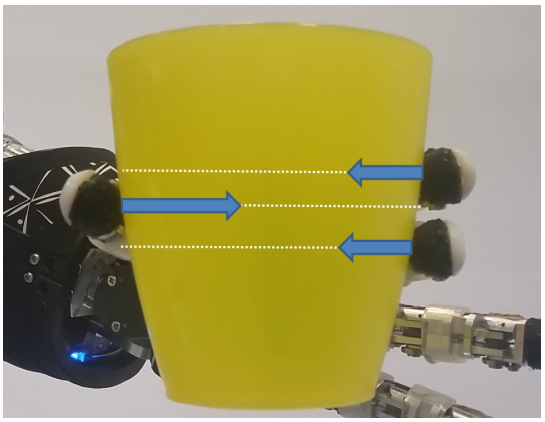


FIGURE 2.2: When the grasp is stable the force directions are assumed to be parallel.

$$\hat{g}(\mathbf{f}) = \underset{g}{\operatorname{argmax}}\, p(g | f_{th}, f_{IM}), \tag{2.2}$$

where for sake of simplicity we omit random variables in the notation. Using Bayes' rule, and assuming $p(g)$ to be uniformly distributed, we can equivalently maximize the likelihood function:

$$\hat{g}(\mathbf{f}) = \underset{g}{\operatorname{argmax}}\, p(f_{th}, f_{IM}|g). \tag{2.3}$$

According to our error model, we can rewrite $\hat{g}(\mathbf{f})$ as follows:

$$\begin{aligned}
\hat{g}(\mathbf{f}) &= \underset{g}{\operatorname{argmax}}(p(f_{th}, f_{IM}|g)) \\
&= \underset{g}{\operatorname{argmax}}\,(p(f_{th}|g) \cdot p(f_{IM}|g)) \\
&= \underset{g}{\operatorname{argmax}}\left(k'(\sigma)e^{-(f_{th}-g)^2/2\sigma^2} \cdot k''(\sigma)e^{-(f_{IM}-g)^2/4\sigma^2}\right) \\
&= \underset{g}{\operatorname{argmax}}\left(e^{-(f_{th}-g)^2/2\sigma^2-(f_{IM}-g)^2/4\sigma^2}\right) \\
&= \underset{g}{\operatorname{argmax}}\left(-\frac{(f_{th}-g)^2}{2\sigma^2} - \frac{(f_{IM}-g)^2}{4\sigma^2}\right) \\
&= \underset{g}{\operatorname{argmin}}\left(2(f_{th}-g)^2 + (f_{IM}-g)^2\right) \\
&= \frac{2}{3} \cdot f_{th} + \frac{1}{3} \cdot (f_{ind} + f_{mid}),
\end{aligned} \tag{2.4}$$

where $k'(\sigma)$ and $k''(\sigma)$ are quantities unrelated to g. This result points out how the estimate f_{th} is more reliable than f_{IM}. This is because f_{IM} sums up noise affecting both the index and the middle fingertips.

2.2.2 System components: force controllers

As first step we developed a framework made of a PID force controller for each finger (Fig. 2.3). The input to the system is the voltage v to the motor actuating the proximal joints, while the feedback is the tactile readings at the fingertip. The other joints (i.e. those actuating the distal and abduction movements) are controlled independently in position. The main difficulties faced to achieve a stable controller lied in the inherent non-linearity of the system, mostly due to the non-negligible friction phenomena, slack effect of tendons and saturation of control commands and/or feedback signals. In order to find the optimal PID gains for the controller, we proceeded as follows:

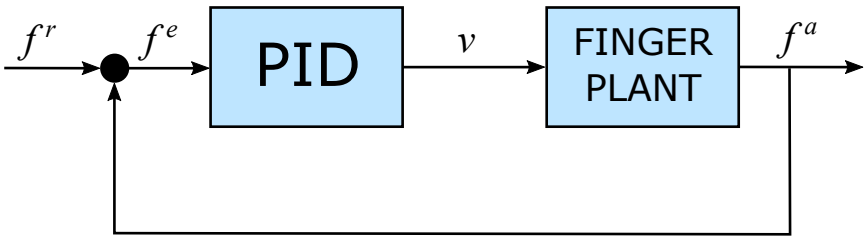


FIGURE 2.3: Force control schema: f^r represents the tactile reference, while f^a is the tactile readings at the fingertip.

- We identified the system by placing the index fingertip in constant contact with an object and by applying step-wise input voltage while measuring the profile of tactile feedback. We executed several experimental sessions varying the initial tactile values, the height of the voltage step and the object used.
- As output of the identification process, we obtained a set of transfer functions $G_i(s)$ describing the system, each one related to a different experimental session.
- We finally used the *Robust Control Toolbox* of MATLAB to identify, using a description of the transfer functions as input, the gains of a controller that is stable in all working conditions.

All the experiments related to the system identification are explained in detail in sec. 2.3.1.

2.2.3 System components: high-level controller

This layer coordinates the fingers by sending proper tactile references to the low-level controllers, with the aim of stabilizing the grasp while maintaining a given grip strength. The controller enforces the constraint in (2.4) and modulates the forces to control the position of the object with respect to a frame attached to the hand. We define the object center, C_o, as the centroid of the triangle identified by the three points of contact between the fingertips and the object (Fig. 2.4). Instead of controlling C_o in the three-dimensional space, we simplify the problem by considering that:

- at this stage, only proximal joints are free to move;
- in an anthropomorphic hand the rotational axis of the proximal joints are nearly parallel to each other.

As a result, C_o ends up moving along a curve path when the object is controlled by our system, which is, in turn, responsible for regulating the final position of the fingers in contact. In order to locate C_o along this path, we define the object position, α_o, as the angle shown in Fig. 2.4. For our purposes, we are not interested in controlling any possible rotation of the object around C_o. For this reason we always set the tactile references of the index and the middle fingers equal.

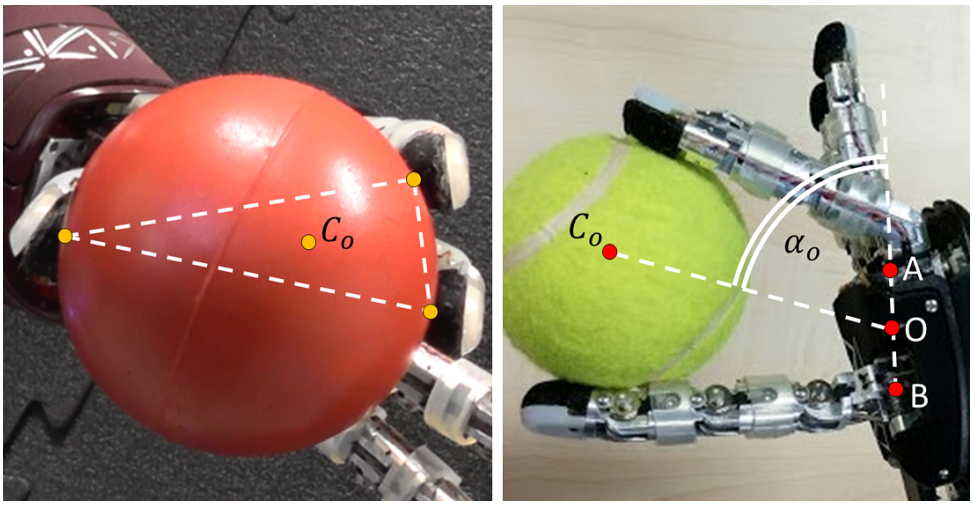


FIGURE 2.4: The object center, C_o, is defined as the centroid of the triangle identified by the three points of contact (left). The object position, α_o, is defined as the angle between the vectors $\vec{OC_o}$ and $\vec{OA}$ (right). A and B are set at the base of the middle finger and the thumb, respectively, while O lies at midpoint between A and B.

The controller objective is to compensate the error between the reference object position, α_o^r, and the actual object position, α_o^a. In order to overcome such an error we use a further PID controller dealing with the following quantity:

$$u(\mathbf{f}) = f_{ind} + f_{mid} - f_{th}. \tag{2.5}$$

The quantity $u(\mathbf{f})$ represents our estimate of the resultant force applied to the object by the fingers. Ideally, for $u > 0$, the object will move towards the thumb (i.e. $\dot{\alpha_o} > 0$), whereas for $u < 0$ the object will move towards the index and the middle fingers (i.e. $\dot{\alpha_o} < 0$); in practice, the equilibrium will be satisfied for $u \neq 0$. As depicted in Fig. 2.5, the high-level PID controller takes the object position error $\alpha_o^e = \alpha_o^r - \alpha_o^a$ as input signal and yields suitable values of the control signal u to be partitioned among the three fingers force controllers with the goal of driving α_o^e to zero. Such a control partition is found as follows: once a specific equilibrium index u^* is requested, and

given a desired grip strength g^*, the set of tactile references to be sent to the underlying low level controllers can be calculated by solving the following system of equations:

$$\begin{cases} u(\mathbf{f}) = f_{ind} + f_{mid} - f_{th} = u^* \\ g(\mathbf{f}) = \frac{2}{3} \cdot f_{th} + \frac{1}{3} \cdot (f_{ind} + f_{mid}) = g^* \cdot \\ f_{ind} = f_{mid} \end{cases} \tag{2.6}$$

The first two equations lead to:

$$f_{th} = u^* - f_{ind} - f_{mid} = \frac{3}{2} \cdot g^* - \frac{f_{ind}}{2} - \frac{f_{mid}}{2}, \tag{2.7}$$

then using the third equation and solving with respect to f_{ind} and f_{mid} we obtain:

$$f_{ind} = f_{mid} = \frac{g^*}{2} + \frac{u^*}{3}, \tag{2.8}$$

and replacing f_{ind} and f_{mid} in equation 2.7 we finally have:

$$f_{th} = g^* - \frac{u^*}{3}. \tag{2.9}$$

The resulting control schema is shown in Fig. 2.5.

2.2.4 System components: stable pose learning

At the top layer in the hierarchy a GMM provides the values of α_o^* and the remaining joints of the hand, $\mathbf{\Theta}_{\mathbf{np}}^*$, that lead to the best grasp in terms of stability, i.e. the grasp that is most likely to prevent the oPbject from slipping, even when external perturbations are applied. The GMM learns a probabilistic model of a set of stable grasp poses. This model is trained by demonstration, i.e., a human operator marks static hand poses related to stable grasps, avoiding grasp configurations that are likely to cause object instability, such as a nonzero momentum applied by the fingers. In addition, to facilitate consequent manipulation tasks, we considered preferable grasp configurations that are far from joint limits and in which the contact points were as close as possible to the center of the fingertips. The number of objects used to perform this training must be sufficient to be able to generalize over unknown objects.

The features that we chose as variables of the GMM are the set $\mathbf{\Theta_{np}}$, the object position α_o and the set $\mathbf{L}$ of lengths of the edges of the triangle defined by the points of contact (Fig. 2.4). We indicate this set of features used to train the model as $\mathbf{G} \equiv < \mathbf{\Theta_{np}}, \alpha_o, \mathbf{L} >$. The likelihood of a given grasp configuration $\mathbf{G}^*$ under a GMM Ω with m components is calculated as follows:

$$p(\mathbf{G}^*|\Omega) = \sum_{i=1}^{m} \pi_i \mathcal{N}(\mathbf{G}^*|\mu_i, \mathbf{\Sigma_i}) \tag{2.10}$$

where π_i is the prior of the i^{th} Gaussian component and μ_i and $\mathbf{\Sigma_i}$ are its related mean and variance; $p(\mathbf{G}^*|\Omega)$ can be considered as a measure of likelihood of stability.

We further characterize $\mathbf{G}$ as $\mathbf{G} = \mathbf{Q} \cup \mathbf{R}$, that is, the union of two subsets. The first, denoted by $\mathbf{Q} = \{\mathbf{L}\}$, contains features that encode the structure of the object and barely change while manipulation takes place. As a result, they cannot be controlled. The second, denoted by $\mathbf{R} = \{\mathbf{\Theta_{np}}, \alpha_o\}$, contains all features that can be controlled.

The main idea is that, given a grasp on an object and its corresponding set of features, $\mathbf{Q}^*$, the model can be exploited in order to infer a set of features $\mathbf{R}^*$ that maximizes $p(\mathbf{G}|\Omega)$, and, consequently, makes the grasp as firm as possible. The features $\mathbf{R}^*$ can be easily retrieved by taking the expectation over the conditional distribution $p(\mathbf{R}|\mathbf{Q}^*, \Omega)$, which can be expressed in closed form:

$$p(\mathbf{R}|\mathbf{Q}^*, \Omega) = \sum_{i=1}^{m} h_i(\mu_{R,i} + \mathbf{\Sigma_{QR,i}}\mathbf{\Sigma_{QQ,i}^{-1}}(\mathbf{Q}^* - \mu_{Q,i})) \tag{2.11}$$

where

$$h_i = \frac{\pi_i \mathcal{N}(\mathbf{Q}^*|\mu_{Q,i}, \mathbf{\Sigma_{QQ,i}})}{\sum_{i=1}^{m} \pi_i \mathcal{N}(\mathbf{Q}^*|\mu_{Q,j}, \mathbf{\Sigma_{QQ,j}})} \tag{2.12}$$

and $\mu_{R,i}$, $\mu_{Q,i}$, $\mathbf{\Sigma_{QR,i}}$ and $\mathbf{\Sigma_{QQ,i}}$ are subparts of the mean μ_i and variance $\mathbf{\Sigma_i}$ as follows:

$$\mu_i = \begin{bmatrix} \mu_{Q,i} \\ \mu_{R,i} \end{bmatrix}, \mathbf{\Sigma_i} = \begin{bmatrix} \mathbf{\Sigma_{QQ,i}} & \mathbf{\Sigma_{QR,i}} \\ \mathbf{\Sigma_{RQ,i}} & \mathbf{\Sigma_{RR,i}} \end{bmatrix} \tag{2.13}$$

Once $\mathbf{R}^*$ is given, we know where to steer the proximal joints of the hand using the high-level controller, whereas the remaining joints are controlled in position to reach their set-point $\mathbf{\Theta_{np}^*}$. In Fig. 2.5 is reported the final control schema which includes the GMM regression.

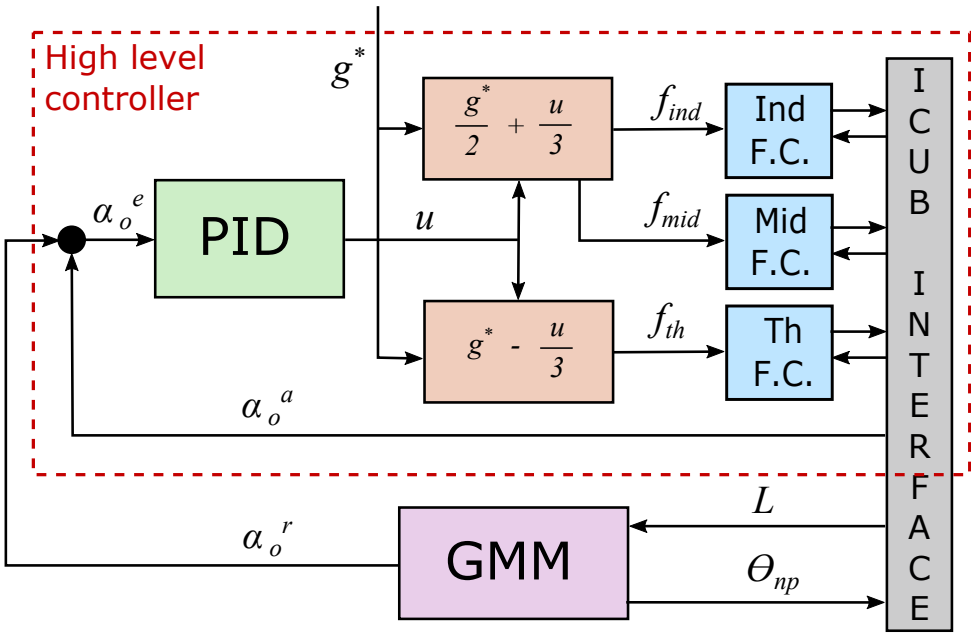


FIGURE 2.5: Final control schema including all the components of our method. The cyan boxes represent the force controllers, while the red dashed line identifies the high-level controller.

2.3 Experiments

In the following, we first describe in details the experiments needed to achieve a robust force controller framework (sec. 2.3.1), then we illustrate the experiments carried out to validate our grasp stabilization method and the related results (sec. 2.3.2).

2.3.1 Experiments: force controllers

In order to tackle the issues related to the non-linearity of the system we started from a simple characterization of the tactile response with resort to classical linear modeling. To develop a robust controller, able to work with objects made of different material, for our experiments we considered two different scenarios. In the first scenario we characterized the tactile response when dealing with hard objects. Specifically, the initial setup consisted in a metal support (see Fig. 2.6) stabilized by sergeants, with the iCub index fingertip in constant contact with its surface. Conversely, in the second scenario we characterized the tactile response for soft objects. In this case, as initial setup we let the iCub hold a soft ball between its thumb and index fingertip, with the thumb locked in position. In both scenarios, as input signal we used voltage applied to the DC motor that actuates through a tendon the index finger proximal joint.

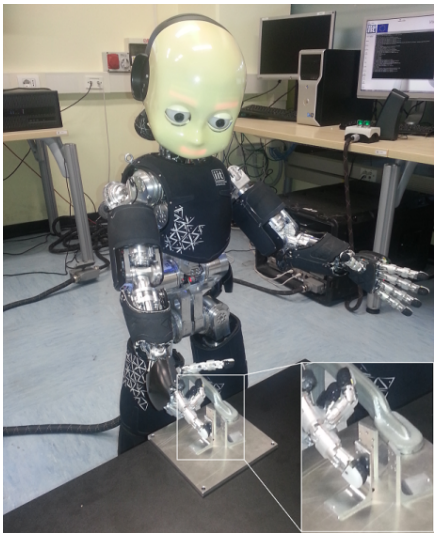


FIGURE 2.6: The setup used to characterize the tactile response for hard objects.

We carried out the system identification with a twofold aim: (1) to verify if under the same starting conditions (i.e. same initial tactile feedback and same input voltage applied to the motor), the system reacts producing repeatable tactile responses, that is equivalent responses in stochastic terms; (2) to identify zones where the system responses can be estimated by means of linear processes that can be profitably employed in the control design. During the open-loop identification we apply step-wise input voltage to the motor actuating the finger and measure back the profile of tactile feedback. In order to better characterize the plant in diverse working zones where the presence of non-linearities affect in different way the overall performance, we vary the initial tactile values and the amount of the voltage step. Once the data acquisitions is complete, we identify the system response in terms of variations, accounting for how the tactile feedback evolves relatively to its starting values.

In the following we report on the four distinct cases of plant responses to positive (step-up) and negative (step-down) step-wise voltage input in both of the above mentioned scenarios. In each case we carried out 8 experiments of 6 acquisitions each.

Step-up identification - hard object

To further limit the effect of uncertainties during the identification, we kept the fingertip always in contact with the metal support. Table 2.1 resumes the data of the 8 experiments, in which as already mentioned we vary the initial tactile values and the amount of the voltage step.

Exp.	Starting taxel response [-]	Starting input voltage [ticks]	Applied step-up voltage [ticks]	No. of repetitions
1	45	300	300	6
2	85	600	300	6
3	110	900	300	6
4	50	300	600	6
5	90	600	600	6
6	60	400	300	6
7	95	700	300	6
8	60	400	600	6

TABLE 2.1: Summary of experiments for step-up identification using a hard object.

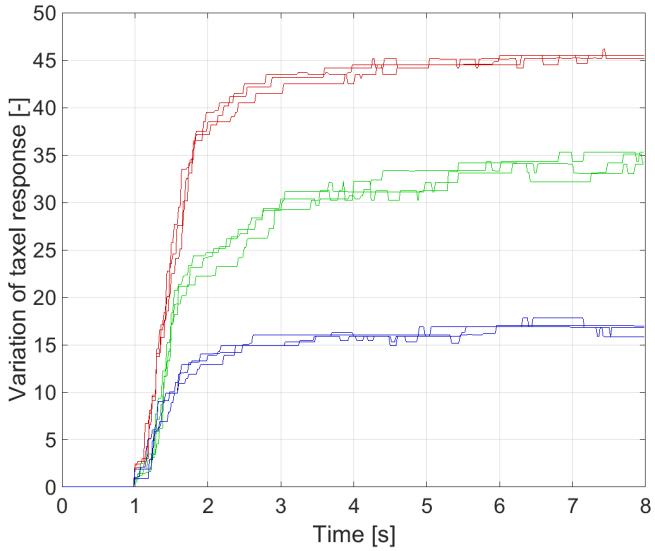


FIGURE 2.7: Plant responses for a positive step-wise voltage input applied at t=1 s using a hard object, for the experiments #6 (green), #7 (blue) and #8 (red) summarized in Table 2.1.

For the repeatability analysis, we show in Fig. 2.7 how the data collected for Experiment #6, #7 and #8 outlined in Table 2.1 indicate a somewhat deterministic behavior of the system (noise apart) throughout the 6 repetitions when solicited always in the same conditions. Nonetheless, the presence of nonlinearities comes evident since the variations in the responses do not reflect the same proportionality seen in the variations of the step-up input voltage.

For the same experiments illustrated above, we then performed an identification stage using the *System Identification Toolbox* of MATLAB. The visual

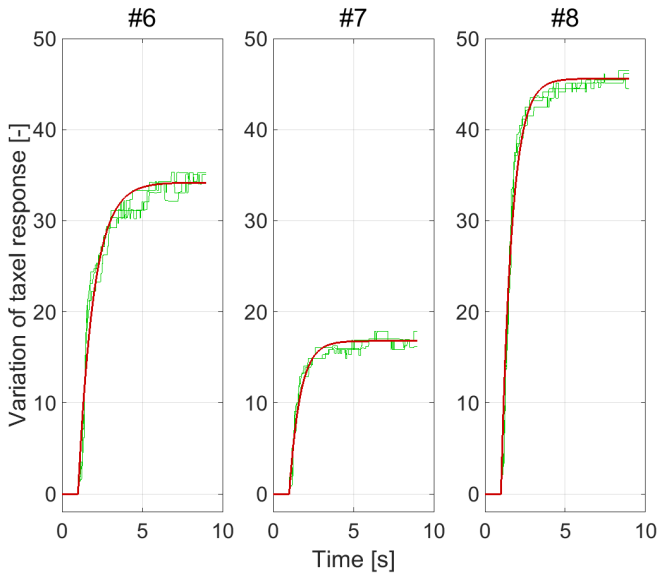


FIGURE 2.8: Tactile predictions (red) given by the identified models along with the last three responses (green) of the experiments #6 (left), #7 (middle), and #8 (right) of the step-up identification using a hard object.

Exp.	K [1/tick]	τ_p [s]
6	0.114	0.963
7	0.056	0.670
8	0.076	0.668

TABLE 2.2: Model parameters of the step-up identification using a hard object.

inspection of the system responses exposing a profile that settles exponentially towards the target does suggest to rely on a very simple process template G identified by the time constant τ_p and the DC gain K. In formulas:

$$G(s) = \frac{K}{\tau_p \cdot s + 1} \tag{2.14}$$

We therefore seek for a convenient set of parameters (K,τ_p) that lets the process G best approximate the acquired tactile output when provided with the step-up voltage as input. The parameters found by the identification process are reported in Table 2.2, where the first 3 responses of the corresponding experiments have been used for the identification. Finally, Fig. 2.8 shows the comparison between the variations in the tactile response predicted by the identified models and the last three repetitions of the experiments, pointing out a good level of accordance with the data.

Exp.	Starting taxel response [-]	Starting input voltage [ticks]	Applied step-down voltage [ticks]	No. of repetitions
1	110	1000	-1000	6
2	125	1000	-1050	6
3	125	1000	-1080	6
4	120	1000	-1100	6
5	75	500	600	6
6	80	500	300	6
7	80	500	300	6
8	80	500	600	6

TABLE 2.3: Summary of experiments for step-down identification using a hard object.

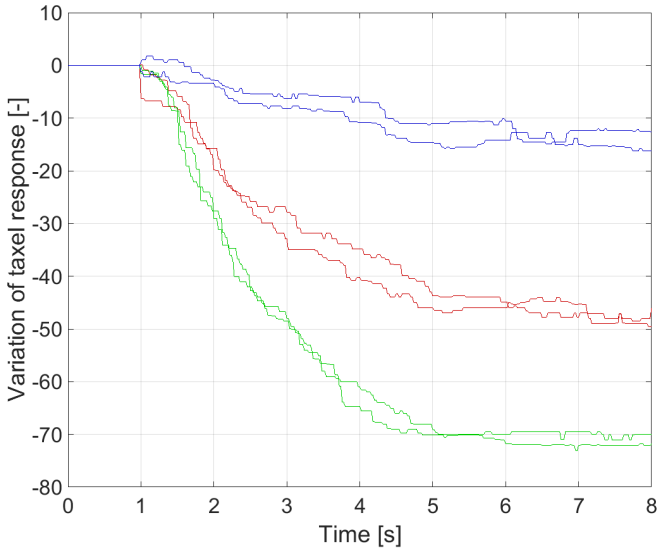


FIGURE 2.9: Plant responses for a negative step-wise voltage input applied at t=1 s using a hard object, for the experiments #2 (red), #3 (green) and #5 (blue) summarized in Table 2.3.

Step-down identification - hard object

Similarly, we run the same identification process also for negative step-wise voltage inputs, still using the metal support, collecting the data corresponding to the 8 experiments described in Table 2.3. As shown in Fig. 2.9, the repeatability of the system output is ensured also in this context, and it comes out that the model template that best fits the system response is still the one obtained for the step-up case.

The values of the time constant τ_p and the DC gain K are summarized in Table 2.4 for the experiments #2, #3 and #5, while Fig. 2.10 validates the plants described by such parameters. We notice that the tactile response converges considerably more slowly with respect to the step-up case, which is

Exp.	K [1/tick]	τ_p [s]
2	0.047	2.06
3	0.067	1.67
5	0.032	3.12

TABLE 2.4: Model parameters of the step-down identification using a hard object.

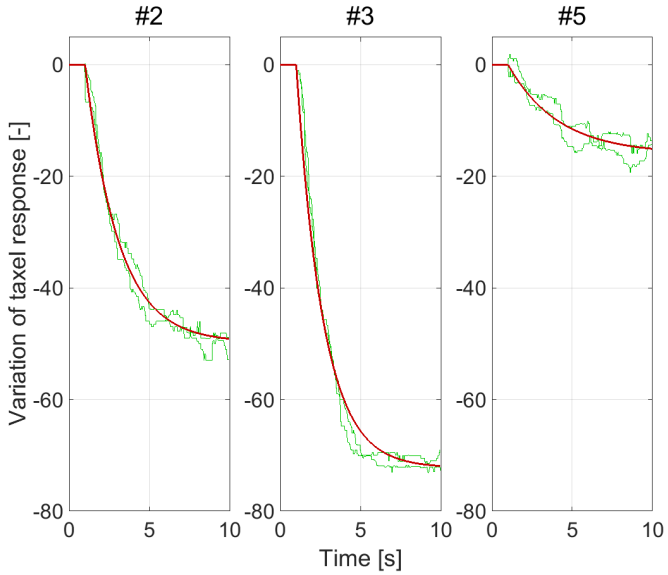


FIGURE 2.10: Tactile predictions (red) given by the identified models along with the last three responses (green) of the experiments #2 (left), #3 (middle), and #5 (right) of the step-down identification using a hard object.

also suggested by the higher values of the τ_p parameters. This is due to complex friction phenomena which leads to a different behavior depending on whether positive or negative voltage input is applied.

Step-up identification - soft object

In the soft object scenario the thumb and the index finger of the iCub are kept constantly in contact with the soft ball. The thumb is locked in position and it is simply used as a support. Table 2.5 resumes the data of the 8 experiments, in which the starting input voltage and the applied step-up voltage are the same as for the hard object/step-up case. However, we see that the starting taxel response is generally different, even if the same input voltage is applied. This is because a soft object is more compliant with respect to a hard object and the surface in contact with the fingertip is much wider, which causes a different taxel activation.

Exp.	Starting taxel response [-]	Starting input voltage [ticks]	Applied step-up voltage [ticks]	No. of repetitions
1	55	300	300	6
2	125	600	300	6
3	190	900	300	6
4	50	300	600	6
5	120	600	600	6
6	75	400	300	6
7	150	700	300	6
8	75	400	600	6

TABLE 2.5: Summary of experiments for step-up identification using a soft object.

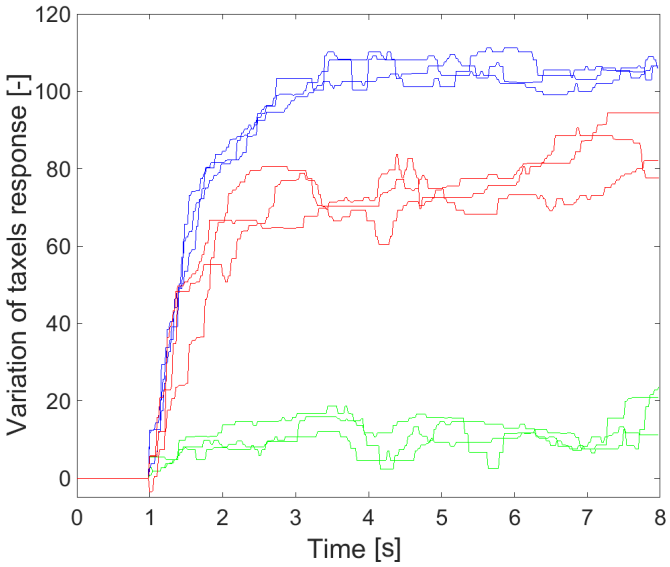


FIGURE 2.11: Plant responses for a positive step-wise voltage input applied at t=1 s using a soft object, for the experiments #3 (green), #4 (blue) and #5 (red) summarized in Table 2.5.

The repeatability of the tactile response is still preserved, as shown in Fig. 2.11 for the experiments #3, #4 and #5 summarized in Table 2.5, but this time the feedback is considerably more disturbed. This is still related to the greater compliance of the material. Indeed, after the application of the step-up voltage the contact surface between the object and the fingertip may vary (along with the number of taxel activations) while compressing the object.

However, the process template that best describes the system output is still the same of equation 2.14, as validated in Fig. 2.12, which show the plants described by the parameters found by the identification process and reported in Table 2.6.

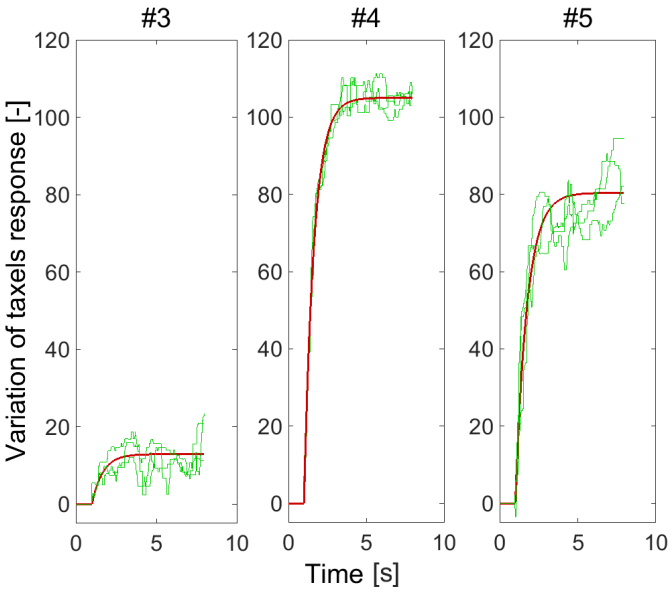


FIGURE 2.12: Tactile predictions (red) given by the identified models along with the last three responses (green) of the experiments #2 (left), #3 (middle), and #5 (right) of the step-down identification using a soft object.

Exp.	K [1/tick]	τ_p [s]
3	0.043	0.673
4	0.175	0.635
5	0.134	0.748

TABLE 2.6: Model parameters of the step-down identification using a soft object.

Step-down identification - soft object

Finally, we run the identification process with the soft object setup but applying negative step-wise voltage inputs. The data related to the experiments is reported in Table 2.7, while Fig. 2.13 shows the tactile responses for experiments #2, #3 and #5. As expected, repeatability is maintained and the same model template is observed. Fig. 2.14 shows the plants validation while Table 2.8 contains the related plant parameters.

Moreover, consistently with previous cases, the tactile response appears to be relatively disturbed (soft object setup) and relatively slow (step-down voltage applied).

Robust PID controller

Overall, we approximated the response of the system in the different conditions with a set of 32 first order $G_i(s)$ systems spanned by pairs of time

Exp.	Starting taxel response [-]	Starting input voltage [ticks]	Applied step-down voltage [ticks]	No. of repetitions
1	170	1000	-1000	6
2	165	1000	-1050	6
3	170	1000	-1080	6
4	175	1000	-1100	6
5	90	500	600	6
6	100	500	300	6
7	100	500	300	6
8	95	500	600	6

TABLE 2.7: Summary of experiments for step-down identification using a soft object.

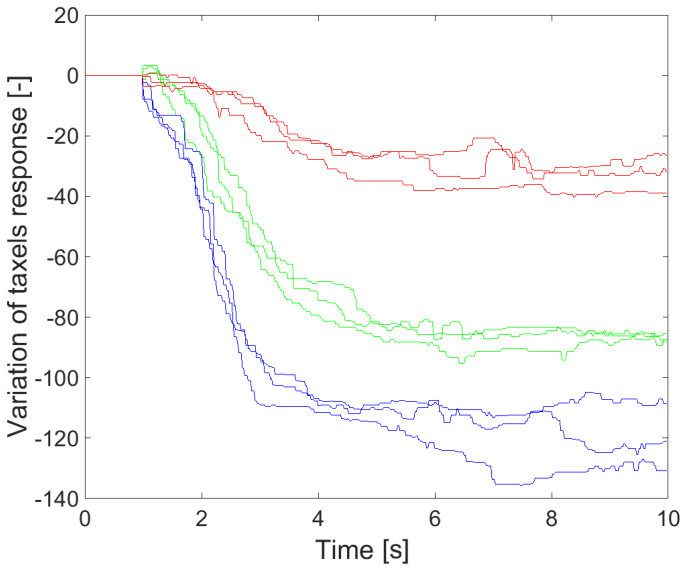


FIGURE 2.13: Plant responses for a negative step-wise voltage input applied at t=1 s using a soft object, for the experiments #2 (red), #3 (green) and #5 (blue) summarized in Table 2.7.

Exp.	K [1/tick]	τ_p [s]
2	0.087	2.16
3	0.113	1.62
5	0.075	3.30

TABLE 2.8: Model parameters of the step-down identification using a soft object.

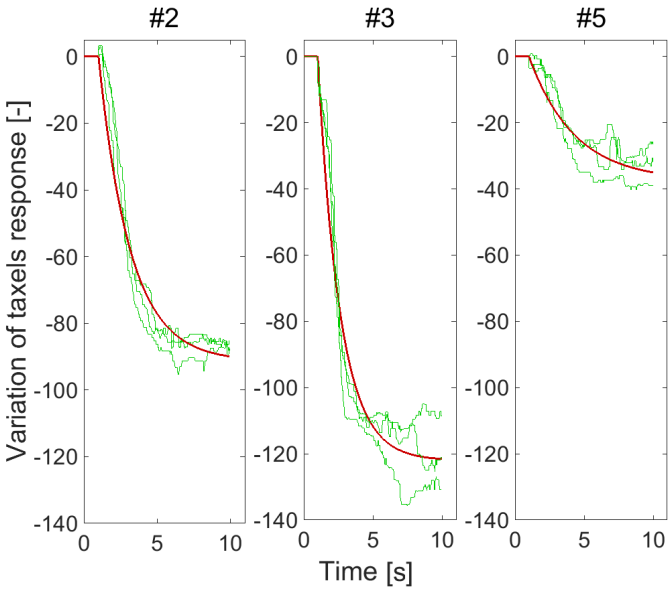


FIGURE 2.14: Tactile predictions (red) given by the identified models along with the last three responses (green) of the experiments #2 (left), #3 (middle), and #5 (right) of the step-up identification using a soft object.

constants $\tau_{p,i}$ and DC gains K_i. Then, as final step we used the *Robust Control Toolbox* of MATLAB to compute, using all the pairs $(K_i, \tau_{p,i})$ as input, the optimal PID gains for the controller (see Fig 2.3 for the control schema). This allowed us to obtain the gains that perform best (in terms of step response characteristics such as rise time, settling time and overshoot) in all the working conditions. Fig. 2.15 shows the simulation of the normalized closed-loop response of the different plants identified during the experiments when a given force reference is set and the optimal PID gains are adopted. Finally, Fig. 2.16 reports the performance of the actual controller while grasping a paper cup and tracking a multi-step force reference.

2.3.2 Experiments: high-level controller and stable pose learning

In order to train the GMM we used a set of 10 objects of different size, shape and material (Fig. 2.17). For each object we carried out 6 different grasps, each starting from a different hand pose. The object position, α_o, and all the joint values in $\mathbf{\Theta_{np}}$ were chosen manually in order to find a pose that was visually determined to be stable under the action of the high-level controller. After each grasp we stored the model features $\mathbf{G}$. The 60 feature vectors were used to train the GMM.

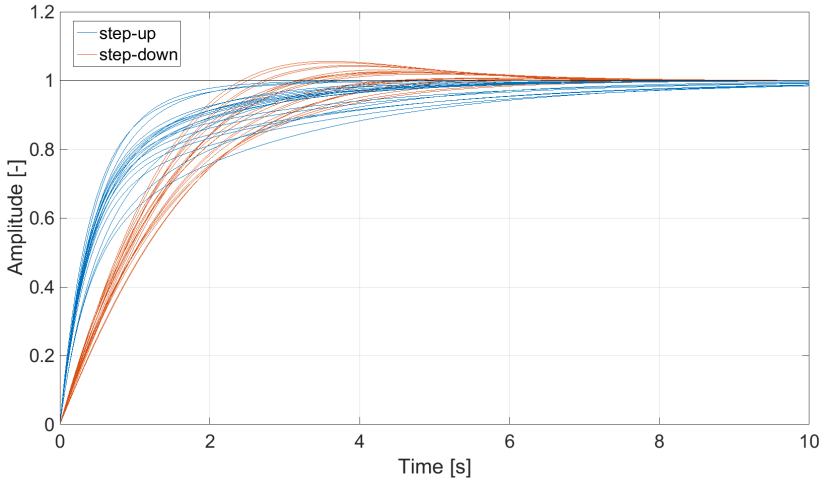


FIGURE 2.15: Simulation of the normalized closed-loop response of the plants identified during the experiments using the optimal PID gains obtained with the *Robust Control Toolbox* of MATLAB.

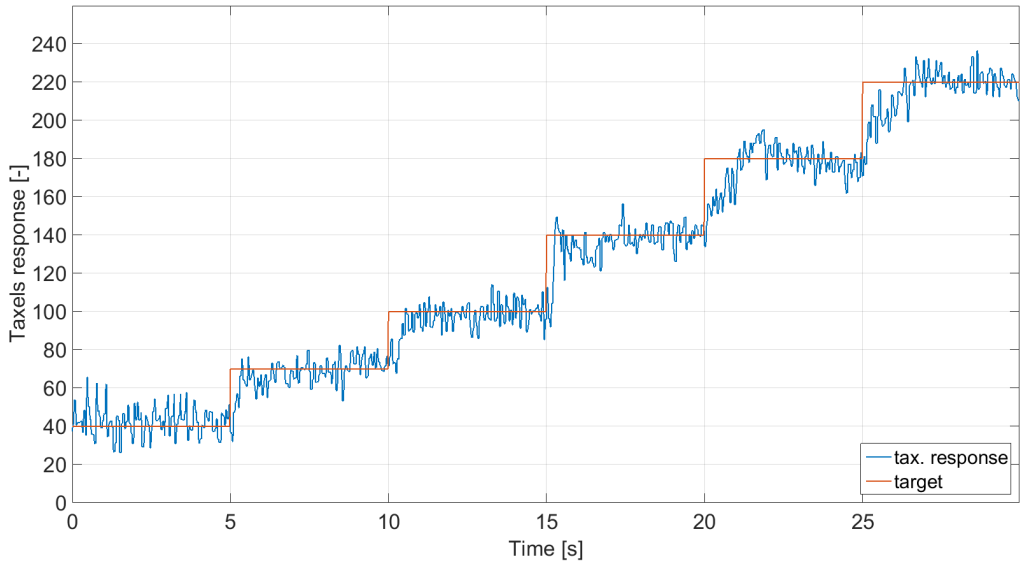
During the training process we explored as much as possible the space **L** of the distances between the points of contact. In this way, the GMM regression becomes reliable and robust with respect to the query point **Q**. In our experiments, the number of Gaussian components, m, is set to 2 using the *Bayesian information criterion*. Before the training process, data was normalized to zero mean with range [-1, 1].

We run several experiments in order to show the effectiveness of our method. We evaluated the tracking performance of the high-level controller in terms of object position and grip strength control and the performance of the grasping adaptation in terms of grasp stability and pose quality.

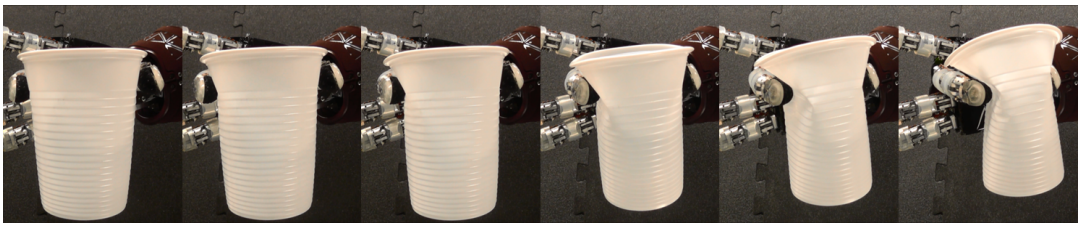
In each experiment we used the four objects shown in Fig. 2.18. These objects were chosen because they have different size, shape and are made of different material. In addition they are different from the ones used for the PID force controller tuning and from the ones used for the GMM training. During the experiments the encoders and the tactile data were sampled at 50 Hz.

Object position tracking

For each object we used the high-level controller to control the object position by tracking sine wave reference signals. Since the GMM regression is not used in this experiment, only proximal joints are controlled while the other



(A)



(B)

FIGURE 2.16: The PID force controller in action while grasping a paper cup with a multi-step force reference which increases every 5 seconds. The reference is sent only to the index fingertip, while the thumb is controlled in position and serves as a support. The figure shows the tactile response observed while tracking the reference (A) and the object snapshots taken after each force increment (B).

FIGURE 2.17: The objects used to train the GMM.

FIGURE 2.18: The objects used to validate the grasp stabilization method.

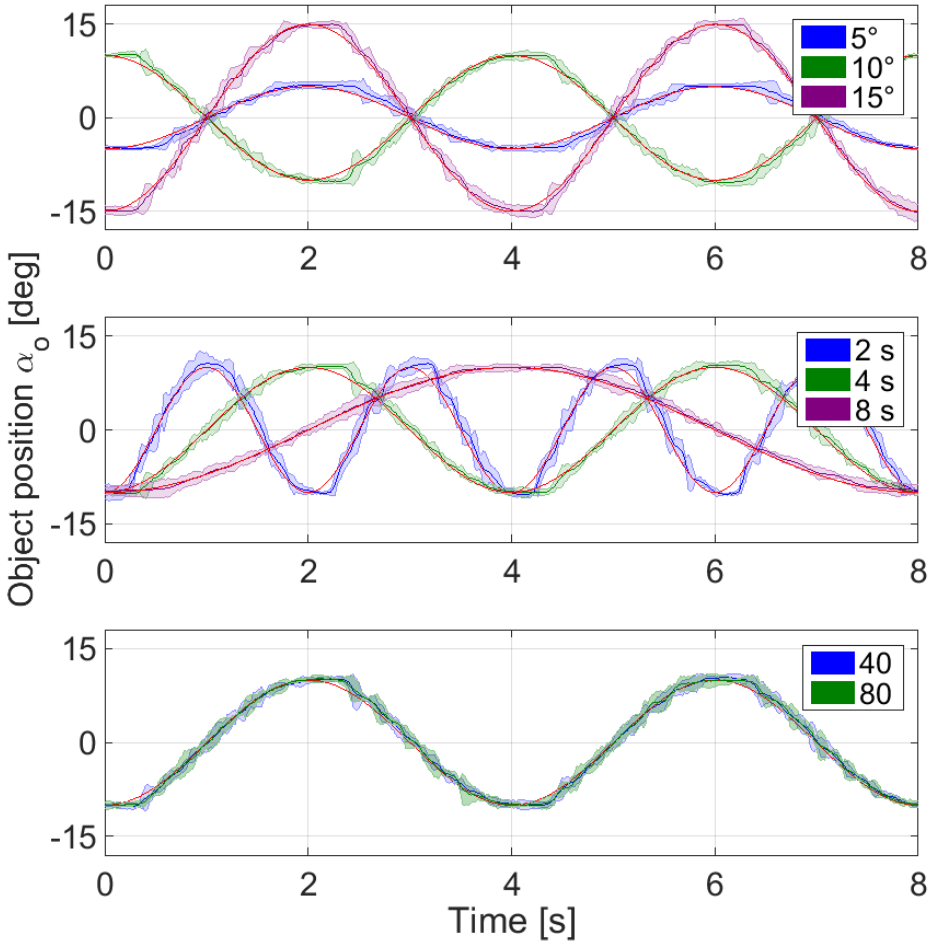


FIGURE 2.19: Object position tracking performance. For each setting, the target object position (in red) and the mean and the confidence interval at 95% over the different objects of the actual object position are shown. We run different trials where we varied the sine wave amplitude (top), period (middle) and the grip strength (bottom) starting from a reference sine wave with amplitude 10°, period 4 sec. and grip strength 80.

joints are maintained fixed at a constant position. During the experiments both the target and the actual object position were collected.

Fig. 2.19 shows the results for all the wave signals considered. In each experiment the controller was able to track the reference reliably and with low error.

Grip strength control

We evaluate the performance of the controller to maintain a desired grip strength on the object. We compare our approach against a baseline, simpler strategy in which the fingers move towards the object at constant speed and stop as soon as contact is detected on the fingertips. The grip strength on the object is measured using the tactile sensors and compared against the grip strength obtained with our controller.

To evaluate our approach for each object, we tried to reach two grip strength references, namely 40 and 80. For the baseline, in order to achieve a given grip strength x, we simply set the force thresholds to the fingers, that is x to the thumb, and $\frac{x}{2}$ to both the index and the middle fingers. For each combination of object and grip strength reference we run 5 trials of the experiment, starting from a different hand pose.

In Fig. 2.20 are shown the results of the experiment. Notice that the grip strength achieved by the baseline is generally higher than the target and quite unpredictable (i.e. it is affected by high variance). Overshooting is probably due to a delay in the tactile response, showing that a proper force control is needed. By contrast, our method is able to maintain the desired grip strength with accuracy, independently from the object and from the target reference.

Grasp stability

To measure the grasp stability we perturbed the hand after the grasp adaptation obtained using the GMM regression model. In particular, we carried out 5 experiments per object and then compared the results with the same baseline controller used for the previous experiment. The perturbation consisted in shaking the hand by means of a sine wave signal sent as position reference to one of the wrist joints. The sine wave had a period of 0.5 seconds, an amplitude of 5 degrees and a total duration of 1 second. In order to make the comparison fair, the same series of hand starting poses was used for both methods under analysis. As measure of stability we counted the number of times in which the object did not fall as a result of the perturbation.

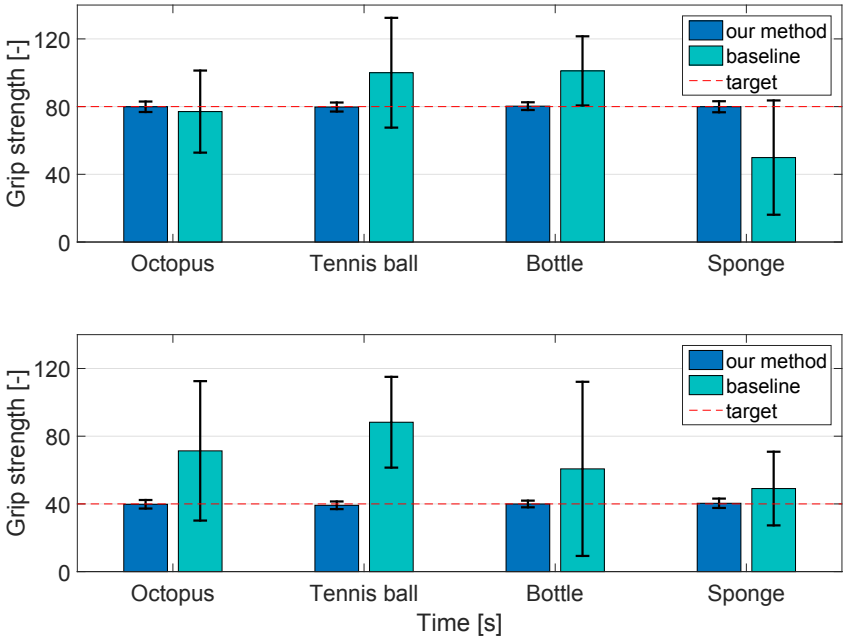


FIGURE 2.20: Grip strength control. The mean and the confidence interval at 95% over the different trials are shown. Notice that our approach allows controlling the grip strength with higher realiability and accuracy than the baseline.

Success rate	Octopus	Tennis ball	Bottle	Sponge
Our method	5/5	4/5	5/5	5/5
Baseline	4/5	1/5	2/5	4/5

TABLE 2.9: Results of the stability performance experiment. For each object the table shows the number of times that the object did not fall after the perturbation.

Table 2.9 shows that grasps obtained with our method were robust to perturbations, even when dealing with slippery objects, like the tennis ball and the bottle. In contrast, the baseline method was unreliable with these objects, meaning that the re-grasp strategy was effective. On the other hand, the performance of both methods on soft objects is similar, since these objects hardly slip independently of the hand configuration.

Hand pose quality

With this experiment we demonstrate that the re-grasp strategy provided by the GMM leads to hand configurations that are preferable for manipulation purposes. Indeed, when grasp adaptation is not triggered, the fingers might keep a configuration that is close to the joint bounds, or in which contact between the object and the fingertips is close to the borders of the fingertip.

In such cases, any consequent manipulation would be limited. In order to quantitatively describe this limiting condition, we introduced the following indexes:

- The *bounds* penalty index:

$$\eta_B = \sum_{i \in \mathbf{\Theta}} \frac{1}{\Theta_i - \Theta_{min\,i}} + \frac{1}{\Theta_{max\,i} - \Theta_i}, \tag{2.15}$$

  where $\mathbf{\Theta}$ is the set of hand's joints, while $\Theta_{min\,i}$ and $\Theta_{max\,i}$ represent lower and upper bounds of the joint Θ_i;

- the *contact* penalty index:

$$\eta_C = \sqrt{d_{th}^2 + d_{ind}^2 + d_{mid}^2}, \tag{2.16}$$

  where d_i is the distance in length between the point of contact on the fingertip i and its center.

For each object of the set, we carried out 5 grasps, changing each time the starting hand pose.

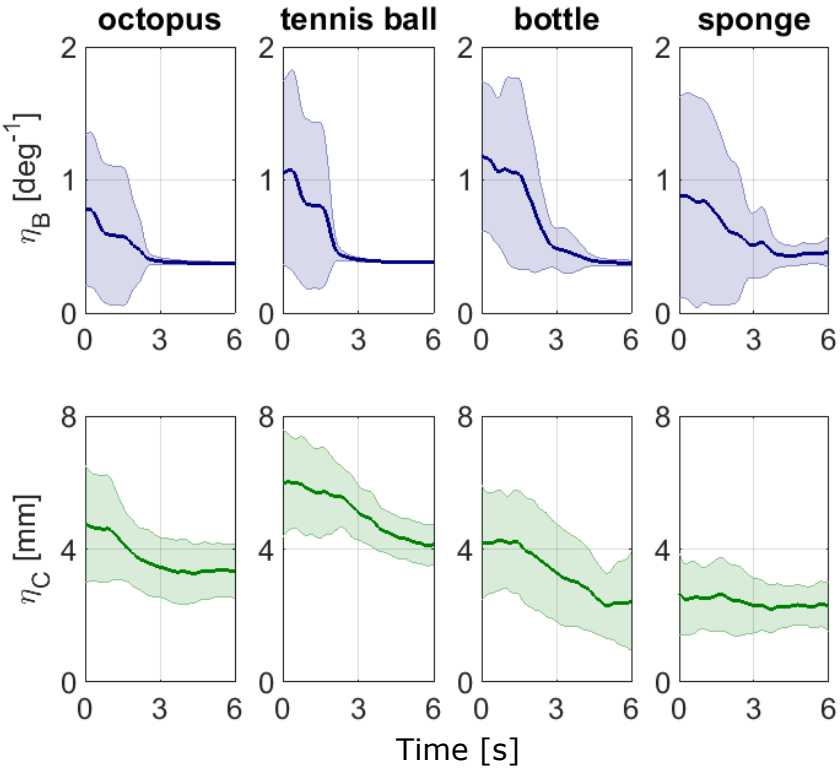


FIGURE 2.21: Hand configuration penalties. The Fig. shows, for each object, the mean and the confidence interval at 95% over the 5 grasps of the penalty indexes.

Fig. 2.21 shows for each object the evolution of the penalty indexes while grasping adaptation is applied. The plot proves that the GMM regression actually manages to move the joints far from their bounds and the points of contact close to the center of the fingertips. In the case of the sponge, η_C does not decrease considerably; this happens because the initial hand poses had already a good configuration. The variance related to η_B strongly reduces over time for all the objects. This makes the method robust with respect to the initial hand pose. In the case of η_C, such effect is much lower, since the points of contact configuration is strongly dependent on the shape of the object and can be hard to control.

2.4 Applications

Our method has been extended and applied in different ways to achieve more complex objectives than just grasp stabilization. In the following sub-sections we show how it has been applied to perform the bi-manual object handover task (sec. 2.4.1), and how it has been extended to achieve object rotation (sec. 2.4.2), while we dedicate a full chapter to explain in detail how we took advantage of the stabilization method properties to carry out tactile object recognition (ch. 3).

2.4.1 Applications: bi-manual object handover

The grasp stabilization method has been exploited to carry out the bi-manual object handover, i.e. the task of passing objects from one hand to the other. A stable grasp is essential for the correct execution of the entire task, especially because the handover method assumes that the object does not move with respect to the hand that holds it. My contribution in this work was to ensure the reliability of the grasp stabilization as required from the task. In the following sub-sections we give an overview of the pipeline adopted by the method and we briefly show the results obtained.

Pipeline overview

Basically, the robot is asked to pass a known object from one hand (that we refer to as *first hand*) to the other hand (*second hand*). The entire pipeline (outlined in Fig. 2.22) can be divided in the following steps:

- *Stable grasp with the first hand:* the robot grasps the desired object with the first hand and controls the grasp using tactile feedback. Grasp stabilization with tactile feedback is active during the entire execution of the task.
- *Point cloud acquisition and filtering:* the robot vision system provides 3D points of the closest blob in the field of view. Then, the point cloud is properly filtered in order to extract only points belonging to the object surface, discarding instead those points that belong to the background or the hand.
- *In-hand localization:* a localization algorithm estimates the object in-hand pose by using the filtered 3D points.
- *Grasping pose selection:* the object model is a-priori annotated with a set of grasping poses reachable by the second hand. After the object is localized, the candidates are ranked according to the distance from the first hand and the manipulability index of the two-arms kinematic chain. Then, the best pose for performing the handover is selected.
- *Approaching strategy:* both arms move until they reach the selected pose.
- *Stable grasp with the second hand:* the robot grasps the object with the second hand and, once the grasp is stabilized, the first hand releases the object. The bi-manual handover is finally achieved.

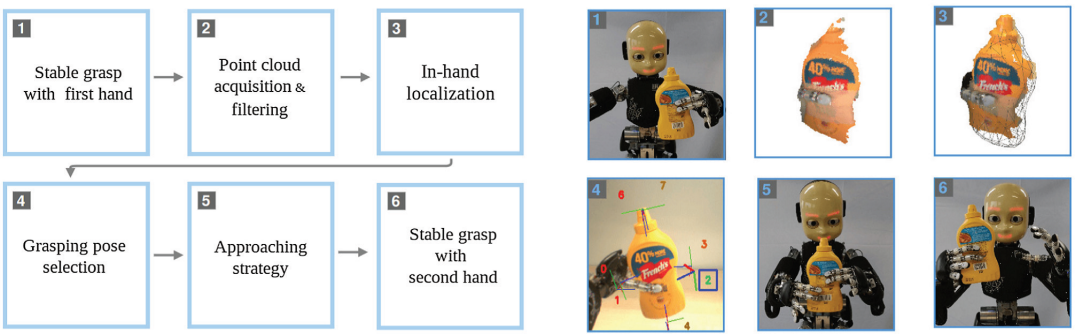


FIGURE 2.22: On the left: a sketch of the pipeline. On the right: some snapshots from the execution of a real handover. (1) The robot grasps the object with the first hand by using tactile stabilization. (2) A set of 3D points of the object is acquired and filtered. (3) The point cloud is used by the localization algorithm for estimating the object pose. (4) The pose for the second hand is selected among a set of previously defined poses. (5) Both arms move so that the second hand achieves the selected pose. (6) Finally, the second hand grasps the object, while the latter is contemporary released by the first hand. Picture from [2].

FIGURE 2.23: The set of objects used in the experiments. We refer to them as: Sugar box, Chocolate box, Mustard box, Chips tube and Little cup. Picture from [2].

It is worth noticing that after the point cloud acquisition, vision is no longer used. This means that during the approaching phase grasp stabilization must prevent the object from moving, because such phase relies on the selected grasping pose, which, in turn, is based on the previously acquired point cloud. Preventing an object from moving, even while the arm itself is moving, is a much stronger condition than just preventing it from falling, so the grasping strategy has to provide a very firm stable grasp.

Moreover, in the last step the second hand manipulates the object while approaching the grasp. Since this happens while the first hand is trying to maintain the grasp stable, the stabilization strategy needs to be very robust to perturbations.

Experiments

The experiments were carried out on the iCub robot using the set of 5 objects shown in Fig. 2.23, by running 10 trials of the method for each object and for different poses. Table 2.10 reports the percentage of success of the handovers (greater than 80% for the majority of the experiments). The task is considered to be successfully achieved if the object is held by the second hand without falling while the arm is moving. Some snapshots of successful handovers are shown in Fig. 2.24.

In order to test the effectiveness of the stable grasp, the handover success rate in presence of stabilization was compared with a baseline obtained in absence of stabilization, i.e. only by closing the fingers until contact is detected on the tactile sensors (Table 2.11). This comparison highlights the effectiveness of the stable grasp since it doubles the success rate of the handover test (from 20–50% of Table 2.11 up to 50–100% of Table 2.10).

Object	Pose	Success rate (%)	Pose	Success rate (%)
Sugar box	Lateral	90	Top	90
Chocolate box	Lateral	90	Top	100
Mustard box	Lateral	80	Bottom	80
Chips tube	Lateral	80		
Little cup	Lateral	50		

TABLE 2.10: Success percentage of the handover task for each object and for different poses. A handover is considered to be successfully achieved if the object is held by the second hand without falling while the second arm is moving.

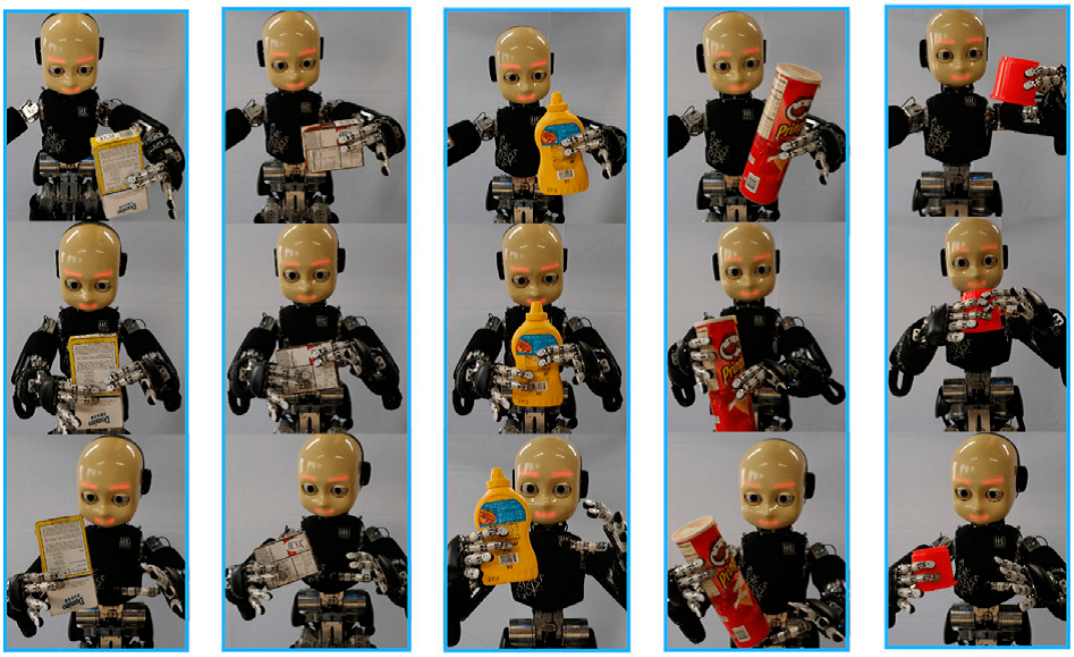


FIGURE 2.24: Examples of successful handovers. Picture from [2].

Object	Pose	Success rate (%)	Pose	Success rate (%)
Sugar box	Lateral	50	Top	50
Chocolate box	Lateral	60	Top	50
Mustard box	Lateral	40	Bottom	30
Chips tube	Lateral	30		
Little cup	Lateral	20		

TABLE 2.11: Success percentage of the handover task for each object and for different poses, in absence of grasp stabilization.

2.4.2 Applications: object rotation

Our stabilization method can be easily extended to perform more complex tasks. Indeed, different GMMs can be trained, still using learning by demonstration, to fit different hand configurations. Then, interesting manipulation tasks can be obtained by switching from a hand configuration to another.

Specifically, we extended our method to achieve object rotation. Object rotation is a very generic task that may turns useful for several purposes, such as pouring liquids, or reaching specific object orientations to facilitate further tasks, such as object handover. Despite its apparent simplicity, object rotation is not generally achieved by humans by simply rotating the wrist, but a complex finger coordination is required in order to also ensure stability. Fortunately, GMMs are ideal to manage such complexity.

Experiments

We proceeded as follows:

- We selected the set of 12 objects shown in Fig. 2.25. They are all different in shape, in order to better generalize over unknown objects.
- For each object in the set, we collected encoders data from 10 stable grasps, 5 in which the object was maintained up-right, and 5 in which the object had been rotated by 45 degrees.
- We used the collected data to train two different GMMs, each one associated to a different object orientation. We refer to the GMM trained while maintaining the object up-right as *first* GMM, while we call the other one *second* GMM.
- We exploited such two models to switch from one hand configuration to the other and carry out object rotation.

Specifically, when an object is given to the iCub, as first step the control schema in Fig. 2.5 is applied using the *first* GMM to achieve a stable grasp with the object maintained up-right. Then, object rotation is carried out by simply replacing the *first* GMM with the *second* GMM. When using the *second* GMM, we applied a little change in the control schema, that is we replaced the set **L** with the hand aperture **A** defined in Fig. 2.26. This is motivated by the fact that, when Gaussian mixture regression is applied, it is preferable that the variable used as input to the controller does not change sensitively during the transition between different hand configurations, because

FIGURE 2.25: The objects used to train the two GMMs related to the object rotation task.

this could lead to instability. When the object is rotated, the relative position between the index and the middle fingers may vary considerably, and as a result **L** would be affected. Conversely, the hand aperture **A**, that is basically related to the size of the object, remains approximately unchanged during the rotation process.

Like for the experiments described in section 2.3, the number of Gaussian components is set to 2 using the *Bayesian information criterion*. Before the training process, data was normalized to zero mean with range [-1, 1].

In Fig. 2.27 we show the 3D points representing the stable grasps used to train the *second* GMM, along with the two Gaussian components obtained from the training process. For visualization purpose, distal joints are not considered in the feature space. The points distribution suggests a strong correlation between the hand aperture **A** and the thumb abduction joint $\mathbf{\Theta_{thA}}$. This is outlined more clearly in Fig. 2.28., which plots the expectation of $\mathbf{\Theta_{thA}}$ over the conditional distribution of **A** according to equation 2.11 and considering $\mathbf{R} = \{\mathbf{\Theta_{thA}}\}$ and $\mathbf{Q} = \{\mathbf{A}\}$. The thumb abduction joint is the joint mostly involved in the rotation process, and the plot suggests that, as expected, the thinner the object (lower hand aperture), the smaller the effort of the joint to perform a 45 degrees object rotation. Finally, Fig. 2.29 show a few examples of successful object rotations.

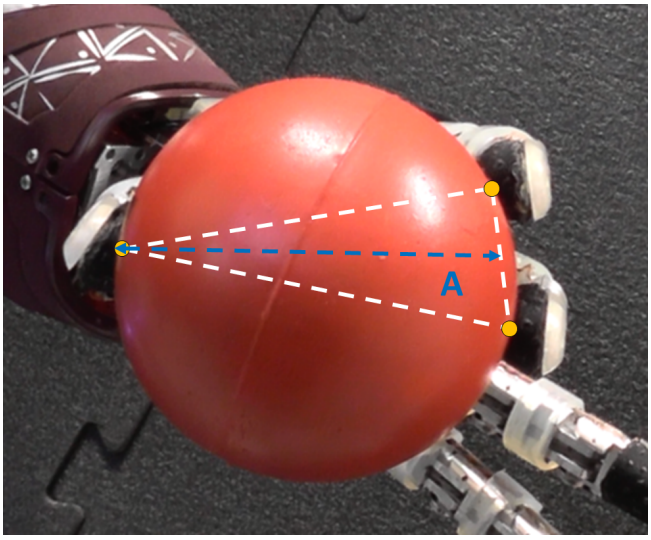


FIGURE 2.26: Hand aperture **A**. Given the triangle identified by the points of contact between the fingertips and the object, the hand aperture is defined as the length of the median related to the side shared by the index and the middle fingers. It is represented by the blue dashed line in the figure.

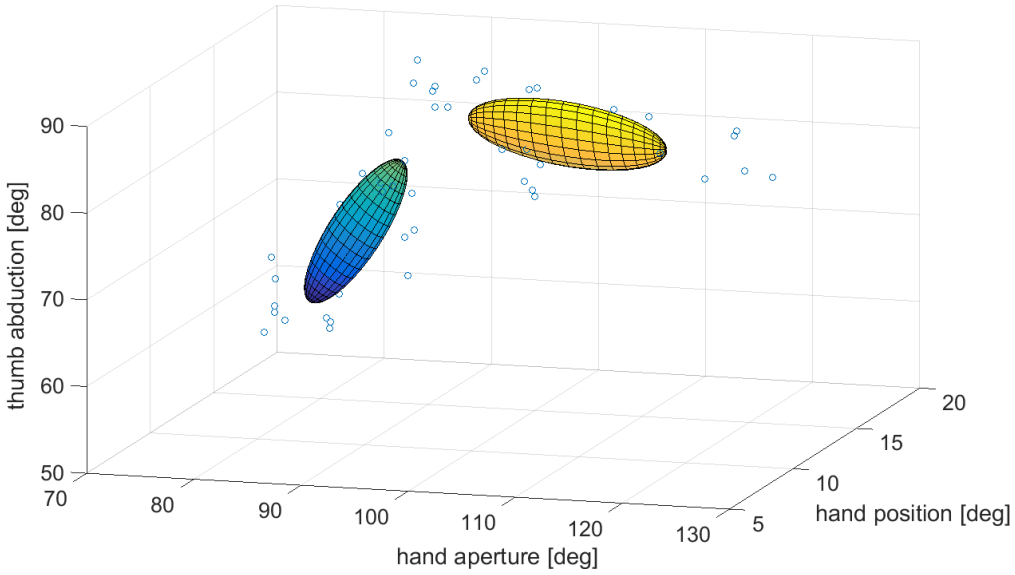


FIGURE 2.27: The figure shows the Gaussian components representing the *second* GMM, along with the 3D points that identify the stable grasps used for the training process.

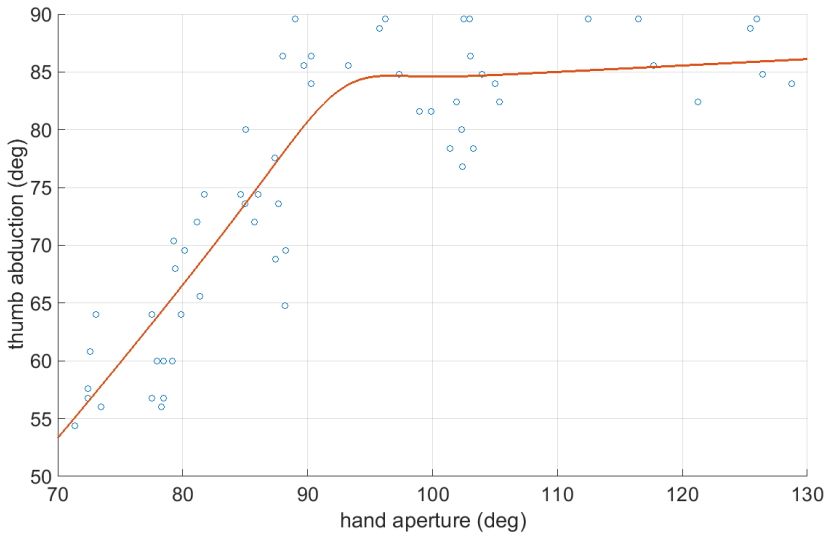


FIGURE 2.28: The figure shows the expectation of the thumb abduction joint over the conditional distribution of the hand aperture in the *second* GMM, along with the 2D points representing the stable grasps used for the training.

FIGURE 2.29: The iCub robot carrying out the object rotation task with 3 different objects. Snapshots are taken at the beginning and at the end of the rotation.

2.5 Conclusion

In this work we dealt with active grasping adaption to unknown objects in order to improve stability. Our method is composed of three components: a low level force controllers framework, a high-level controller that coordinates the fingers to achieve grasp stabilization and grip strength control, and a machine learning approach based on GMM regression aimed at further improving the grasp stability. The method is made simple and effective by taking advantage of the anthropomorphic structure of the hand. Furthermore, since forces are regulated by the high-level controller, the amount of data needed for the GMM training is strongly reduced.

We tested our method on the iCub robot to demonstrate that our approach allows to reliably control the position of the object in the hand, while controlling the grip strength. Finally, we validated the re-grasp strategy provided by the GMM to demonstrate that it allows achieving grasp poses that are preferable for manipulation purposes and robust to perturbations.

The novelty of our work is that it performs grasp adaptation while allowing explicit control on the grip strength on the object using tactile feedback. Such feature can be useful in many applications. For example, it allows to adapt the grip-strenght to avoid slip while handling fragile objects or to squeeze the object to extract material properties for object recognition or subsequent in-hand manipulation. On this regard, in the next chapter we describe in detail how our stabilization method can be exploited to perform tactile object recognition. Furthermore, our method has been exploited to achieve bi-manual object handover and it has been extended using multiple GMMs to perform object rotation.

Chapter 3

TACTILE OBJECT RECOGNITION

3.1 Introduction

In the context of object recognition, tactile sensing provides information that can only partially be acquired by vision. Indeed, properties such as object texture and softness can be better investigated by actively interacting with the object. In order to detect such properties, different approaches have been proposed. Takamuku *et al.* [37] use a tendon driven robotic hand to explore different objects by performing actions such as squeezing and tapping. The collected data is then categorized using Self-Organizing Map. Johansson *et al.* [38] developed a microphone based texture sensor and a hardness sensor that measures the compression of the material at a constant pressure. The texture and hardness data are then merged into one single representation to optimize the object recognition performance.

Psychologists have shown that humans make specific exploratory movements to get cutaneous information from the objects [39], that include, pressure to determinate compliance, lateral sliding movements to determinate surface texture, and static contact to determine thermal properties. Hoelscher *et al.* [40] apply these exploratory movements to a set of 49 different objects grouped in different materials using the BioTac finger. The collected tactile data is analyzed in depth to compare different learning methods, different extracted features and different feature concatenation strategies. In addiction, they also examine classifying objects not by instance, but by material class, to detect the material of a given unknown object.Xu *et al.* [41] use the BioTac finger with the Shadow Dexterous Hand to perform similar movements in order to identify objects by their compliance, texture and thermal properties. When identifying an object, exploratory movements are intelligently selected using a process called Bayes exploration, whereby exploratory movements

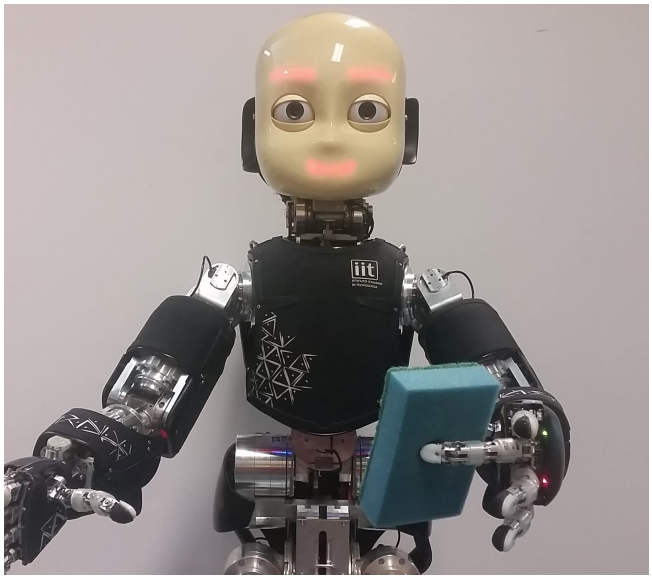


FIGURE 3.1: The iCub humanoid robot carrying out the object recognition task.

that provide the most disambiguation between likely candidates of objects are automatically selected.

All these approaches carry out exploratory movements using a single finger and assume that the object does not move. Conversely, other works recognize an object by grasping the object, putting less restrictions on the hand-object interaction. Schneider *et al.* [9] propose a bag-of-words approach in which each object is grasped several times, learning a vocabulary from the tactile observations. The vocabulary is then used to generate a histogram codebook which is the core identification mechanism. To reduce the number of required grasp actions, they apply a decision-theoretic framework that minimizes the entropy of the probabilistic belief about the type of the object. Chitta *et al.* [42] propose a hybrid velocity-force controller that grasps an object safely and extract at the same time simple features that reveals its deformation properties. They show that a robot can use these features to infer if the object is empty or full and open or close. Chu *et al.* [43] use the PR2 humanoids robot to apply five exploratory procedures to 51 objects that were annotated by human participants with 34 binary adjective labels. Then, they apply static and dynamic machine learning methods in order to find a relationship between such adjectives and the extracted tactile features, so that a humans-like haptic description can be provided also for previously unfelt objects.

Most of these approaches do not deal with the stability problem and assume that the object is laying on, or are fixed to a surface such as a table.

When the object has to be held in the robot's hand, stability problems such as preventing it from falling, make the task of extracting features through interactions more challenging. Kaboli *et al.* [44] employ the Shadow hand to explore the texture properties of various objects while they are held in the hand. This is conducted via the small sliding movements of the fingertips of the robot over the object surface, providing information-rich data used to train classifiers based on different learning algorithms. Gorges *et al.* [45] present an approach to classify an object directly from the haptic sensor data acquired by a palpation sequence with the robot hand. They identify a single palpation step as a description of a finite set of essential finger positions and tactile contact patterns. A palpation sequence can then be merged into a simple statistical description of the object and finally be classified. Higy *et al.* [46] propose a method in which the robot identifies an object by carrying out different exploratory behaviours such as hand closure, and shaking and rotating the object. In their method the authors investigate the role of different sensory modalities and present a strategy to efficiently combine them in a hierarchical classifier to differentiate objects.

In these approaches the stability is typically managed by performing a power grasp, that is, wrapping all the fingers around the object. This means that in general, the final hand configuration after the grasp is not controlled. It strictly depends on the way the object is given to the robot. Due to this, the tactile and proprioceptive feedback suffer from high variability. This requires a larger number of grasps to be performed and negatively affects the performance. Moreover, performing power grasps may limit further actions that could help in extracting other object features such as softness/hardness.

In this work we propose a novel method for in-hand object recognition that uses the controller described in ch. 2 to stabilize a grasped object. The controller is used to reach a stable grasp and reposition the object in a repeatable way.

We perform two exploratory behaviours: squeezing to capture the softness/hardness of the object; and wrapping all of the fingers around the object to get information about its shape. The stable pose achieved is unique given the distance between the points of contact (related to the size of the object), resulting in high repeatability of features, which improves the classification accuracy of the learned models. Differently from other methods, we do not put any restrictions on the objects.

We validated our method on the iCub humanoid robot [11] (Fig. 3.1). We show that using our method we can distinguish 30 objects with 99.0% $\pm$ 0.6%

accuracy. We also present the results of a benchmark experiment in which the grasp stabilization is disabled. We show that the results achieved using our method outperforms the benchmark experiment.

In the next section we present our method for in-hand object recognition. In section 3.3 we describe the experiments carried out to validate our method, while in section 3.4 we present our results. Finally, we conclude in section 3.6 with a discussion.

3.2 Methodology

Here we present the method used to perform the in-hand object recognition task. We use an anthropomorphic hand, but the method can be easily extended to any type of hand that has at least three fingers, two of which are required to be opposed to each other. An important assumption in this work is that the object is given to the robot by a collaborative operator, in such a way that the robot can grasp it by closing the fingers. The remaining steps are performed by the robot autonomously, namely:

- grasping the object using a precision grasp, that is, only the tip of the fingers are in contact with the object,
- reaching an optimal stable pose,
- squeezing the object to get information about its softness,
- wrapping all the fingers around the object to get information about its shape.

We start by explaining how we adapted the grasp stabilization controller to this specific task. This is followed by a description of the feature space, and then we give a brief overview of the machine learning algorithm used to discriminate the objects.

3.2.1 Grasp stabilization

Grasp stabilization is a crucial component of our method for two reasons. First, it is needed to prevent the object from falling, for example, when executing actions like squeezing. Second, reaching a stable and repeatable pose for a given object improves the classifier accuracy. We use the method described in ch. 2 to stabilize the object, with the only difference that the index

finger is not involved in the grasping process. This is done to make it available for the hand enclosure action, with the aim of further enriching the information about the object shape. In order to use two fingers instead of three the grasp stabilization control schema shown in Fig. 2.5 needs to be slightly modified. In the rest of this section we quickly explain what are the main differences with the controller described in the previous chapter. The low-level controllers are not affected by the number of fingers involved in the grasp, so we will discuss only the changes made to the grip strength definition, to the high-level controller schema and to the stable pose learning component.

Grip strength

All the steps described in sec. 2.2.1 can still be applied to the case of two-finger grasps by simply replacing f_{IM}, the sum of the tactile readings at the index and the middle fingertips, with f_{mid}, the tactile reading at the middle fingertip. As a result, we get:

$$\hat{g}(\mathbf{f}) = \underset{g}{\operatorname{argmax}}(p(f_{th}, f_{mid}|g)) \tag{3.1}$$

Moreover, considering that $g = k \cdot F_{th} = k \cdot F_{mid}$ and that $f_i|g \backsim \mathcal{N}(g, \sigma)$, we can rewrite $\hat{g}(\mathbf{f})$ as follows:

$$\begin{aligned}
\hat{g}(\mathbf{f}) &= \underset{g}{\operatorname{argmax}} \left(k'(\sigma) e^{-(f_{th}-g)^2 / 2\sigma^2} \cdot k''(\sigma) e^{-(f_{mid}-g)^2 / 2\sigma^2} \right) \\
&= \underset{g}{\operatorname{argmax}} \left(-\frac{(f_{th}-g)^2}{2\sigma^2} - \frac{(f_{mid}-g)^2}{2\sigma^2} \right) \\
&= \underset{g}{\operatorname{argmin}} \left((f_{th}-g)^2 + (f_{mid}-g)^2 \right) \\
&= \frac{1}{2} \cdot f_{th} + \frac{1}{2} \cdot f_{mid},
\end{aligned} \tag{3.2}$$

where $k'(\sigma)$ and $k''(\sigma)$ are quantities unrelated to g.

High-level controller

The high-level controller can be easily modified to use two fingers considering that:

- the object center C_o can be reasonably defined as the halfway point between the two points of contact (see Fig. 3.2),

- the estimate of the resultant force applied to the object by the fingers is now simply defined as $u(\mathbf{f}) = f_{mid} - f_{th}$,
- the equation involving the index fingertip is no longer needed in the system of equations 2.6, which boils down to:

$$\begin{cases} u(\mathbf{f}) = u^* \\ g(\mathbf{f}) = g^* \end{cases}. \tag{3.3}$$

As a result, the set of tactile references to be sent to the underlying low-level controllers turn out to be the following:

$$f_{mid} = g^* + \frac{u^*}{2}, \; f_{th} = g^* - \frac{u^*}{2}. \tag{3.4}$$

The resulting control schema is shown in Fig. X.

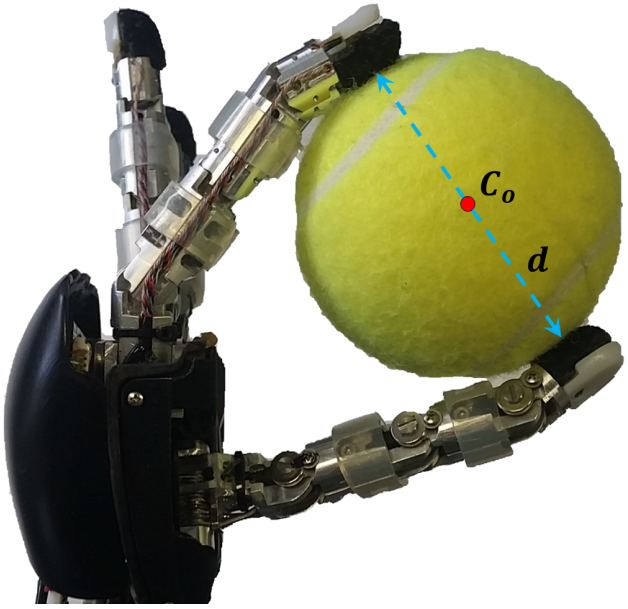


FIGURE 3.2: In two-finger grasps, the object center C_o is defined as the halfway point between the two points of contact.

Stable grasp model

All the concepts exposed in sec. 2.2.4 still holds for the case of two-finger grasps. The only difference lies in the fact that the quantity **L**, which is part of the feature space and represents the set of lengths of the triangle identified by the points of contact in three-finger grasps, is replaced by the quantity d defined as the distance between the two points of contact (see Fig. 3.2).

The set of objects used to train the GMM are still the ones shown in Fig. 2.17, but the training process is entirely repeated performing two-finger grasps, following the same procedure as detailed in sec. 2.3.2. The final control schema of the full grasp stabilization controller is shown in Fig. 3.3.

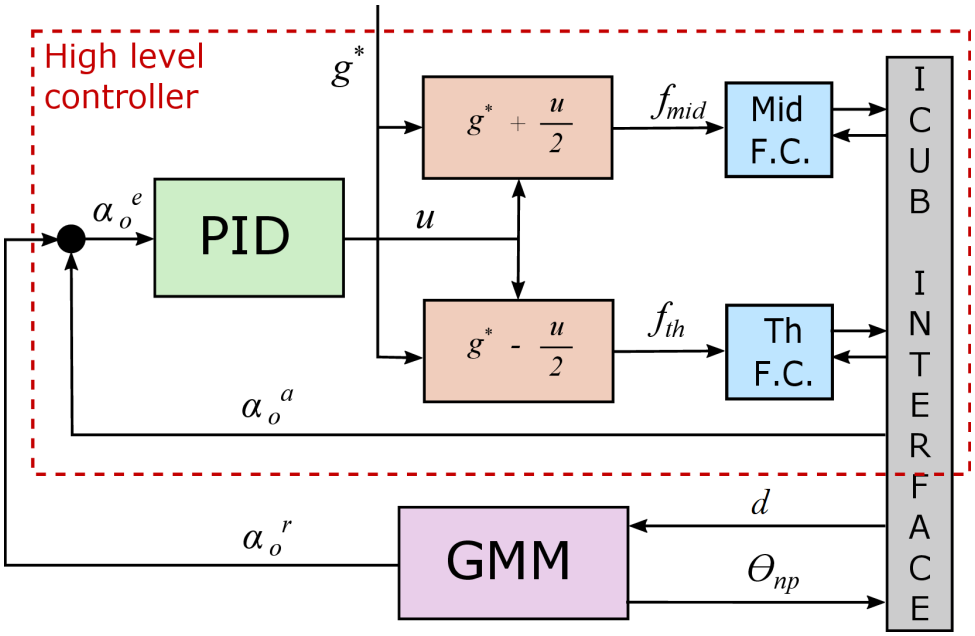


FIGURE 3.3: Final control schema of the grasp stabilization controller used for two-finger grasps. The cyan boxes represent the force controllers, while the red dashed line identifies the high-level controller.

3.2.2 The Feature Space

Once a stable grasp is achieved, the robot manipulates the object to capture its softness and shape by performing two exploratory behaviours: a) squeezing the object between the thumb and the middle finger, and b) wrapping all the fingers around the object. The softness of the object is captured both by the distribution of the forces in the tactile sensor and the deflection of the fingers when the object is squeezed between the fingers of the robot. The shape of the object is captured by wrapping all of the fingers of the robot around it.

As mentioned earlier, the grasp stabilization implies a high degree of repeatability of the achieved pose, independent of the way the object is given to the robot. Thereby, the features produced during the exploratory behaviours exhibit low variance between different grasps of the same object. Which, in turn, increases the accuracy of the classifier.

Tactile responses

The distribution of forces in the tactile sensors is affected by the softness of an object. A hard object will exert forces that are strong and concentrated in a local area. A soft object, in contrast, will conform to the shape of the fingertip and exert forces across all tactile sensors. The tactile sensors also capture information on the local shape of the object at the point of contact. We use the tactile responses from the thumb and the middle finger, $\boldsymbol{\phi}$, in our feature space, since the objects are held between these two fingers.

Finger encoders

The finger encoders are affected by the shape and the harness/softness of the object. When the robot squeezes the object, a hard object will deflect the angles of the finger more than a softer object. Since we use only the thumb and the middle finger during the squeezing action, we use both the initial and the final encoder values for these fingers – $\boldsymbol{\Theta}^{\mathbf{init}}_{\mathbf{grasp}}$ and $\boldsymbol{\Theta}^{\mathbf{fin}}_{\mathbf{grasp}}$, respectively.

To capture the shape of the object, the robot wraps the rest of its fingers around the object. We also include the encoder data, $\boldsymbol{\Theta}_{\mathbf{wrap}}$, of these fingers in our feature space.

3.2.3 The learning algorithm

In order to train the classifier, we used as features the data acquired during the grasping, squeezing and enclosure phase, as described in the previous section. We simply concatenated the collected values, obtaining the feature vector $[\boldsymbol{\Theta}^{\mathbf{init}}_{\mathbf{grasp}}\ \boldsymbol{\Theta}^{\mathbf{fin}}_{\mathbf{grasp}}\ \boldsymbol{\Theta}_{\mathbf{wrap}}\ \boldsymbol{\phi}]$ composed of 45 features, 21 related to the encoders and 24 related to the tactile feedback.

As learning algorithm we adopted Kernel Regularized Least-Squares using the radial basis function kernel. For the implementation we used GURLS [47], a software library for regression and classification based on the Regularized Least Squares loss function.

3.3 Experiments

To test our method, we used the iCub humanoid robot. We used a set of 30 objects shown in Fig. 3.6, of which, 21 were selected from the YCB object and model set [48]. Using a standard set helps in comparing the results of different methods. The objects were selected so that they fit in the iCub

robot's hand without exceeding its payload. The YCB object set did not have many soft objects fitting our criteria, hence, we supplemented the set with 9 additional object with variable degree of softness. We also paid attention to choose objects with similar shape but different softness, as well as objects with similar material but different shapes.

3.3.1 Data collection

The dataset to test our method was collected using the following procedure (depicted in Fig. 3.4):

1. The iCub robot opens all of its fingers.

2. An object is put between the thumb and the middle finger of the robot. The robot starts the approach phase, which consists of closing the thumb and the middle finger until a contact is detected in both fingers. A finger is considered to be in contact with an object when the force estimated at its fingertip exceeds a given threshold. To capture variations in the position and the orientation of the object, each time the object is given to the robot, it is given in a different position and orientation.

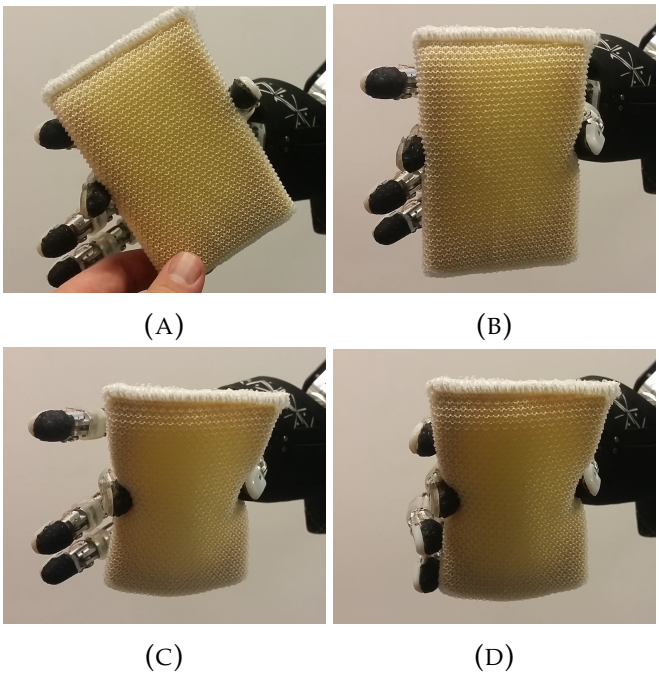
3. At this point the grasp stabilizer is triggered with a given grip strength. The initial value of the grip strength is chosen as the minimum grip strength needed to hold all the objects in the set. The method described in section 3.2.1 is used to improve the grasp stability. When the grasp has been stabilized, the robot stores the initial values of the encoders of the thumb and the middle finger.

4. Then the robot increases the grip strength to squeeze the object and waits for 3 seconds before collecting the tactile data for the thumb and the middle finger. At this point the robot also records the encoder values for the thumb and the middle finger.

5. Finally, the robot closes all of the remaining fingers around the object until all fingers are in contact with the object. At this point, the robot collects the values of the encoders of the fingers.

These steps were repeated 20 times for each object. To test our algorithm we use a fourfold cross-validation. That is, we divide the dataset into 4 sets. We hold one of the sets for testing and use the other three to train a classifier.

This is repeated for all 4 sets. We compute the accuracy and the standard deviation of our classifier using the results of these 4 learned classifiers.



(A) (B)

(C) (D)

FIGURE 3.4: The steps accomplished to identify the object: approach (a), stabilization (b), squeezing (c) and enclosure (d). Notice that the controller repositions the object irrespective of its initial pose. As discussed in the text this greatly improve repeatability and, consequently, recognition performance.

3.3.2 Benchmark experiment

To test our hypothesis that reaching a stable pose improves the classification results we carried out an experiment in which we disable part of the grasp stabilization. As described earlier and depicted in Fig. 3.3, the grasp stabilization consists of three modules: the low-level force controller, the high-level controller and the stable grasp model. We only disable the stable grasp model. The other two components are needed to stop the object from slipping and to control the grip strength.

The stable grasp model produces two outputs: a) the target object position, α_o^r, and the target set of non-proximal joints, $\mathbf{\Theta_{np}}$. In the benchmark experiment we calculate the value of α_o^r and the $\mathbf{\Theta_{np}}$ when the thumb and the middle finger make contact with the object. That is, the alpha is set to the current position of the object and the theta is set to the current joint configuration. Apart from this difference, the high-level controller and the low-level force controller are still active, controlling grip strength and maintaining a

Features	Θ^{init}_{grasp}	Θ_{grasp}	Θ_{all}	ø	All
Mean	80.5%	93.3%	96.3%	95.0%	99.0%
Std	2.0%	0.8%	0.7%	0.8%	0.6%

TABLE 3.1: Classification accuracies using our method with classifiers trained using different set of features.

stable grasp. However, without the stable grasp model, the grasp is less robust to perturbations.

Henceforth, unless stated otherwise, when we mention that the grasp stabilization is disabled, we mean that we only disable the repositioning based on the GMM. Hence, we collected the data for the benchmark experiment following the same steps as described in section 3.3.1 where the grasp stabilization was disabled.

3.4 Results

In this section we present the results of our method and show how each of the selected features in our feature space helps in capturing different properties of the objects, namely, the softness/harness and the shape of the object. This will be followed by a comparison between our method and the benchmark method in which the grasp stabilization is disabled. When reporting the results for brevity we concatenated some of the features: Θ_{grasp}= $[\Theta^{init}_{grasp}\ \Theta^{fin}_{grasp}]$, and Θ_{all} = $[\Theta^{init}_{grasp}\ \Theta^{fin}_{grasp}\ \Theta_{wrap}]$.

3.4.1 Finger encoders

To study the effectiveness of the encoder features, we trained a model using different combinations of these features. Table 3.1 reports the results of these experiments. We notice that using only the initial encoder values, the accuracy is already quite high, 80.5% $\pm$ 2.0%, while including the final encoder values of the thumb and the middle finger after squeezing it increases to 93.3% $\pm$ 0.8%. This is because the fingers will move considerably if the object is soft, thereby, capturing the softness of the object. Fig. 3.6 shows the confusion matrices for the experiments. We notice that several pairs of objects such as the tennis ball (11) and the tea box (30) or the sponge (26) and the soccer ball (28) are sometimes confused if only the initial encoders values are used as features, while they are discriminated after the squeezing action.

Finally we analysed the results of including all encoder data, that is, including the data when the robot wraps its fingers around the object. This improved the classification accuracy to 96.3% $\pm$ 0.7%. From the confusion matrices we notice that adding such features resolves a few ambiguities, such as the one between the soccer ball (28) and the water bottle (22) and the one between the yellow cup (24) and the strawberry Jello box (19). Indeed, these pairs of objects have similar distance between the points of contact when grasped, and cause similar deflections of the fingers when squeezed, but have different shapes.

3.4.2 Tactile responses

As discussed earlier the tactile sensors are useful in capturing the softness of the objects as well as the local shape of the objects. In Fig. 3.5 we can see that using only the tactile feedback we get an accuracy of 95.0% $\pm$ 0.8%, which is comparable with the 96.3% $\pm$ 0.7% obtained using the encoder values. Although they have similar classification accuracy, studying the confusion matrices reveals that objects confused by them are different. For example, the classifier trained using only the tactile data often confuses the Pringles can (1) and the tomato can (7), since they are hard and share similar local shape. Conversely, due to their slightly different size they are always distinguished by the classifier trained using only encoder data. This means that combining the two feature spaces can further improve the accuracy of the learned classifiers.

3.4.3 Combining the two features

Finally, using the complete feature vector we get an accuracy of 99.0% $\pm$ 0.6%. We also notice that the standard deviation in our experiments is decreasing as we add more features. From the confusion matrix we can see that several ambiguities characterizing each individual classifier are now solved. A few objects are still confused due to their similar shape and softness, namely the apple (5) and the orange (6), and the apricot (16) and the prune (10). Less intuitively, the classifier once confuses the apricot with the SPAM can (21), and once it confuses the apricot with the brown block (18). To explain the confusion between these objects, we notice that there is a particular way to grasp them such that the joints configuration is very similar. This happens when the middle finger touches the flat side of the apricot, and the little finger misses both objects.

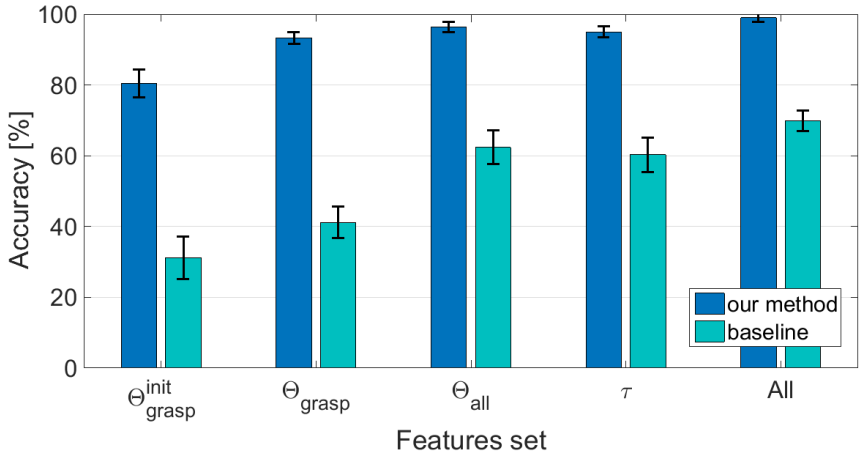


FIGURE 3.5: Summary of the results comparing our method with the benchmark method for different set of features. It shows that our method outperforms the benchmark method with statistical significance. The error bars are standard deviations.

3.4.4 Comparison with the benchmark experiment

Fig. 3.5 shows the results of running the same analysis on the data collected in the benchmark experiment where the grasp stabilization was removed. The results show that the proposed method performs significantly better than the benchmark experiment, achieving 99.0% $\pm$ 0.6%, compared to the benchmark experiment which achieved an accuracy of 69.9% $\pm$ 1.4%. This is because the stabilization method proposed in this paper increases the repeatability of the exploration, which makes the feature space more stable. Indeed, the initial position of the object in the hand strongly affects the collected tactile and encoders data. This variability is reduced using the grasp stability controller. Note that the accuracy of the benchmark experiment increases as more features are added, showing that the feature space is able to capture the object properties.

We run a further analysis to study the effect of increasing the number of trials in the training set. In this case we always trained the classifier with the complete feature vector and considered 5 trials per object for the test set, while we varied the number of trials in the training set between 3 and 15. Fig. 3.7 shows the results of this analysis. The results show that the proposed method boosts the accuracy of the classification, requiring less samples to be able to distinguish the objects. The trend of the accuracy obtained using the benchmark method suggests that it may improve by increasing the number of samples in the training set. However, this is not preferred because it makes

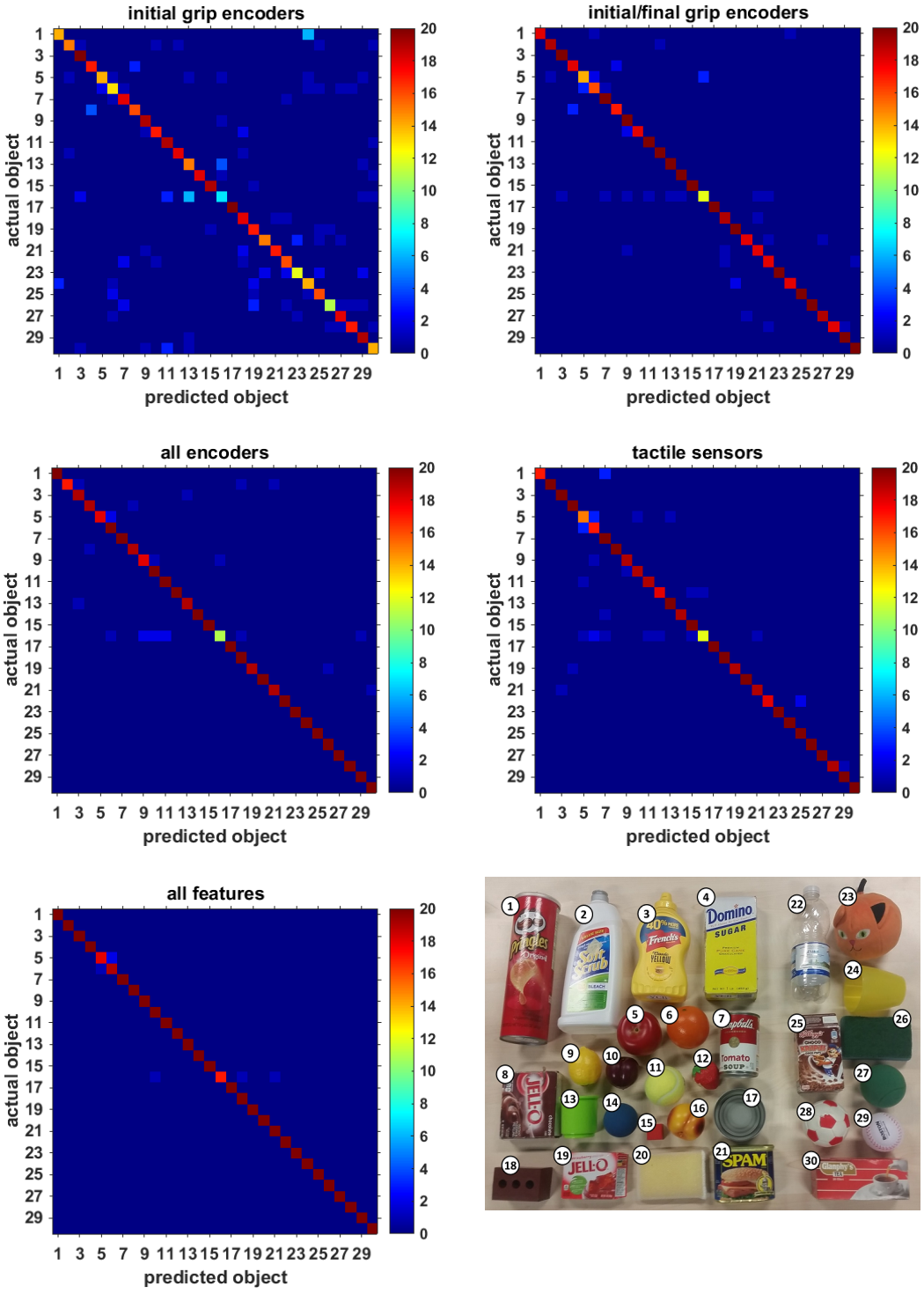


FIGURE 3.6: The confusion matrices obtained using our method with different sets of features. At the bottom right, is the object set used for the experiments. It is composed of 21 objects taken from the YCB object set (left), and additional 9 objects of various degree of softness (right).

Features	Θ^{init}_{grasp}	Θ_{grasp}	Θ_{all}	ø	All
Mean	85.0%	91.4%	95.0%	94.1%	97.6%
Std	3.1%	1.5%	1.8%	1.6%	0.5%

TABLE 3.2: Classification accuracies using our method on the YCB objects only.

it impractical to collect data on large sets of objects, adversely affecting the scalability of the learned classifier.

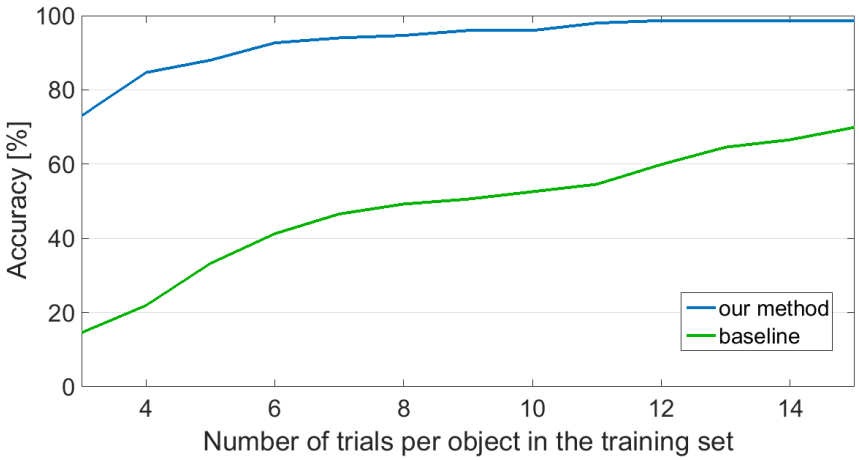


FIGURE 3.7: The accuracy of our method and the benchmark method as a function of the number of training set samples. Our method obtains high accuracy even with a much lower number of training samples.

3.4.5 Results using objects form the YCB set only

Finally, in table 3.2 we provide the results of our method using only the object from the YCB object set, in order to let researchers having the same dataset compare their results with ours.

3.5 Mixing sensory modalities

As mentioned in sec. 3.1, tactile sensing can be used to detect properties such as object texture and softness that are not easily distinguishable using vision. However, it is also true the opposite. Many object properties, such as color or global shape are better captured by vision rather than touch. Therefore, it comes natural to combine the two sensory modalities to produce a classifier able to overcome the limitations of each sensory modality singularly considered.

Specifically, we combined the output of our tactile classifier and the output of a previously developed visual classifier to train a new higher level multimodal classifier that outperforms the others. In the following subsections, we first give an overview of the structure of the visual classifier (sec. 3.5.1), then we show how we combined the output of each classifier to obtain the multimodal classifier (sec. 3.5.2). Finally, we present the procedure followed for the experiments (sec. 3.5.3) and the related results (sec. 3.5.4).

3.5.1 Visual classifier

The visual recognition system running on the iCub robot and used for the experiments is depicted in Fig. 3.8. The system components are related as follows:

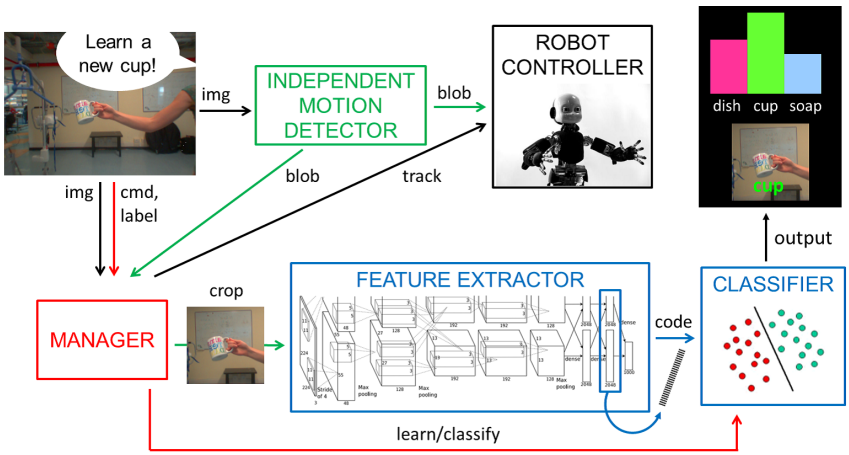


FIGURE 3.8: The visual recognition system of the iCub robot. Picture from [3].

- The independent motion detector component takes as input the images captured by the left and right cameras and create a disparity map to locate the closest object in the scene.
- The object location is used by the robot controller to perform visual tracking, and by the manager component to produce a cropped image in which the useless data around the object is removed.
- The cropped image is processed by the feature extractor component, which provides as output a suitable representation of the object.
- The feature vector is given as input to the classifier component, whose behaviour depends on its internal status (*learn*/*classify*), set by the manager component. If the status is *learn*, the feature vector is simply stored

in memory, to be used afterwards to train a SVM classifier as soon as a sufficient number of feature vectors has been collected. If the status is *classify*, the trained SVM classifier analyzes the feature vector to finally classify the object.

The feature extractor component basically consists in a particular convolutional neural network, called *ResNet-50* [49]. Despite it was originally trained on the ImageNet dataset [50], it turns out that the values computed at the highest layer of the neural network, which are the actual output of the component, provide a very good representation of even unknown objects, being invariant to transformations that do not affect the actual object class and discriminative with respect to other classes.

3.5.2 Multimodal classifier

There are several ways to combine different sensory modalities in order to improve object recognition accuracy. For example, tactile and visual features could be properly mixed to directly train a classifier. Another approach, studied in [46] and followed in this work, is to train separately each monomodal classifier, and then combine their output to finally train a higher level multimodal classifier (see Fig. 3.9). As output of each monomodal classifier we consider the array of real-valued scores assigned to the objects involved in the training process, according to the *one-vs-all* strategy adopted in both learning algorithms. We used different methods to combine the tactile and visual scores, obtaining the following sets of features:

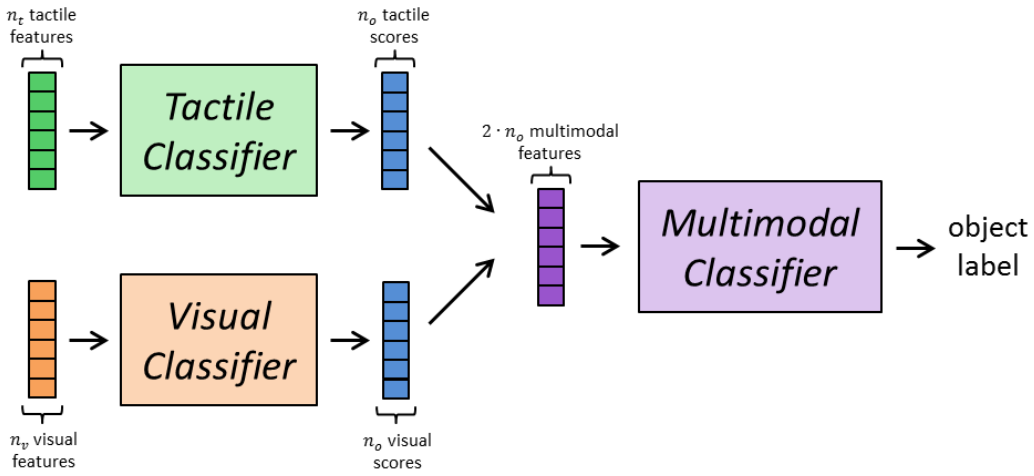


FIGURE 3.9: The pipeline followed to classify objects using the multimodal classifier. n_t, n_v and n_o represent, respectively, the number of tactile features, the number of visual features and the number of objects.

- *normalized* features: the scores are first normalized, independently for each sensory modality, and then concatenated,
- *binary* features: each score is set to +1 if > 0, otherwise it is set to -1, then they are concatenated,
- *normalized-binary* features: starting from the *normalized* features, each score is set to +1 if > 0, otherwise it is set to -1,
- *best-binary* features: the higher score for each sensory modality is set to +1, while the remaining scores are set to -1, then they are concatenated.

These sets of features represent the possible input to the higher level multimodal classifier, which is trained using the learning algorithm Kernel Regularized Least-Squares, already adopted for the tactile classifier (see sec. 3.2.3).

3.5.3 Experiments

For the experiments we used the object shown in Fig. 3.10. We expressly included in the set couples of objects that could be easily confused either by touch or by vision to better emphasize the ability of the multimodal classifier to solve the ambiguities characterizing each monomodal classifier.

FIGURE 3.10: The objects used to train the multimodal classifier. The chocolate boxes appear identical, but one has been emptied in order to be recognized by touch.

In order to let the iCub grasp the object, we followed a different approach than the one described in sec. 3.3.1. Indeed, while previously a human operator had to directly hand the object to the iCub, now we put the object on a table and let the robot grasp it autonomously. This is done to prevent any possible bias introduced by the operator himself. The autonomous grasp strategy is based on the same principles adopted to perform the bi-manual

handover task (see sec. 2.4), with the difference that this time the object to be grasped lies on a table instead of being held in the hand.

More in detail, the data collection used to train and test the multimodal classifier was carried out by performing, for each object, 21 iterations of the following procedure:

- The iCub is set as in Fig. 3.11a, looking at the surface of a table. The arms are raised in order to be excluded from its visual field.
- The object is put on the table, then the vision system exposed in the previous section is used to extract an instance of the visual features. In every iteration we vary the object location and orientation.
- The object blob in the image and the related depth information are used to locate the object in the operative space and compute a suitable approach path to reach the object with the hand.
- The iCub follows the computed approach path and set the hand around the object (see Fig. 3.11b).
- The object is then grasped (see Fig. 3.11c) and its material properties are explored following the procedure described in sec. 3.3.1, providing an istance of the tactile features.

The collected data is then split into three different sets, each one containing 7 instances of tactile features and 7 instances of visual features for each object. The first set is used to train the monomodal classifiers, which subsequently receive the second set as input to produce the tactile and visual confidence scores. Such scores are used to train the multimodal classifier, that is finally tested using the third set. This process is repeated 24 times, splitting each time the original data in different combinations. As a consequence, the results exposed in next section are averaged over such repetitions.

3.5.4 Results

Following the steps exposed above we trained a tactile classifier, a visual classifier, and four multimodal classifiers, that is one for each set of features listed in sec. 3.5.2. Fig. 3.12 reports the confusion matrix of the tactile classifier. Predictably, the purple and the orange bottles are frequently confused with each other, since they share the same shape. Despite the chocolate boxes have also

(A) (B) (C)

FIGURE 3.11: The iCub autonomously grasping an object to extract tactile information. Starting from the initial setup (A) the image of the object captured by the camera and the related depth information are used to locate the object in the operative space. An approach path is then computed and the object is finally reached (B) and grasped (C).

identical shape, the grasping procedure manages to discriminate them most of time, because the empty box slightly warps when squeezed.

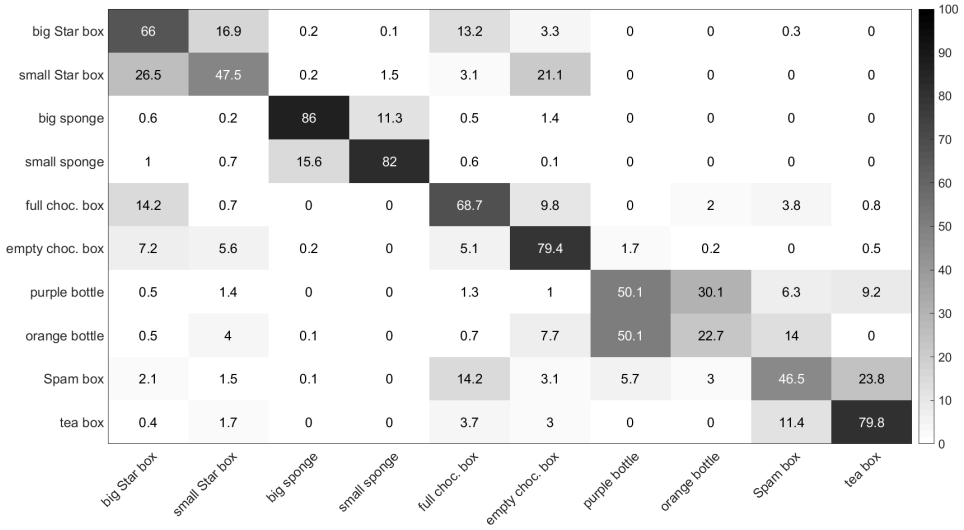


FIGURE 3.12: The confusion matrix of the tactile classifier.

Similarly, Fig. 3.13 shows the confusion matrix of the visual classifier. As expected, it confuses all the couples made of similar objects, i.e. the sponges, the Star boxes and the chocolate boxes. However, differently from the tactile classifier, for each couple the visual classifier always correctly classifies one object, while it misclassifies the other. This is mainly due to the low variance in the visual data used for testing.

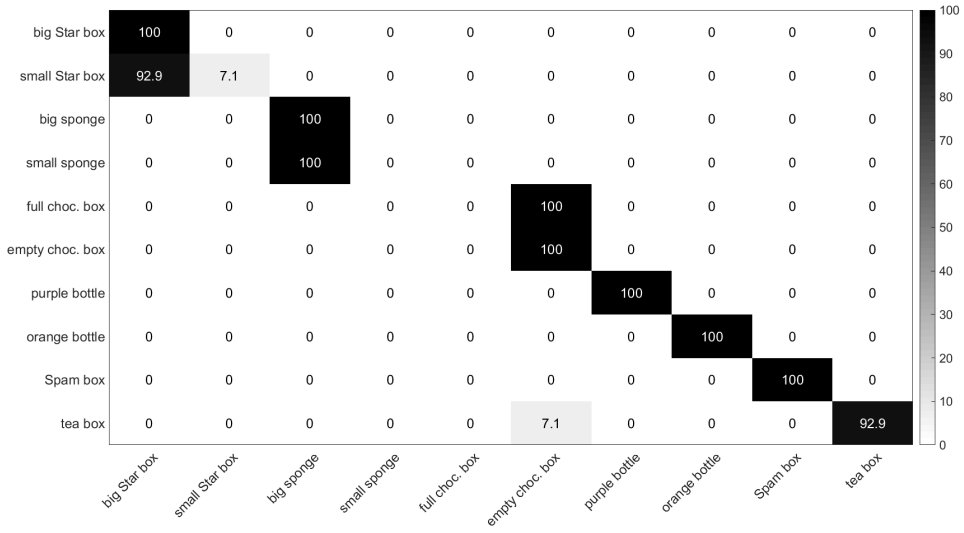


FIGURE 3.13: The confusion matrix of the visual classifier.

In Fig. 3.14 we compare all the accuracies of the monomodal and multimodal classifiers. Differently from the results obtained in [46], where the *binary* classifier got the best result, we achieved the highest accuracy (91.8%) by simply using the *normalized* set of features. In particular, we notice a remarkable improvement with respect to the performance of the individual tactile and visual classifiers, which achieved, respectively, an accuracy of 62.9% and 70.0%. Finally, Fig. 3.15 reports the confusion matrix of the *normalized* multimodal classifier, showing that all the ambiguities observed for the monomodal classifiers are now partially solved.

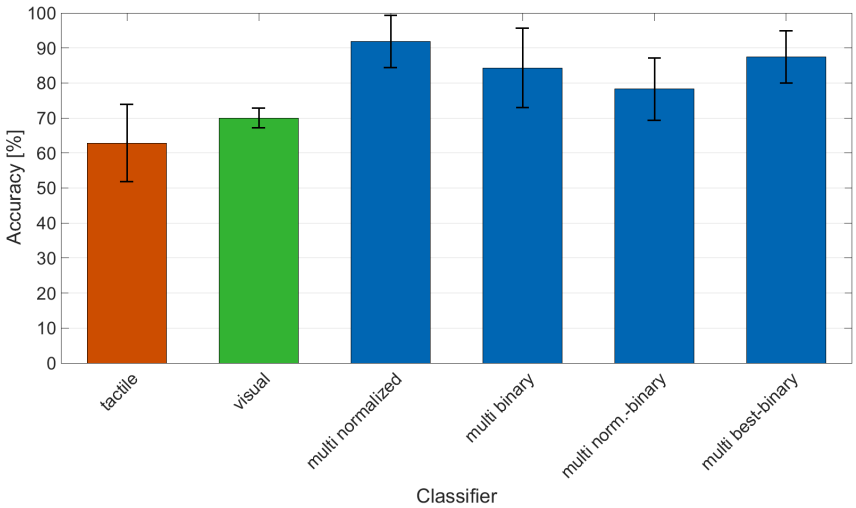


FIGURE 3.14: Comparison of the accuracies achieved by the tactile, visual and multimodal classifiers.

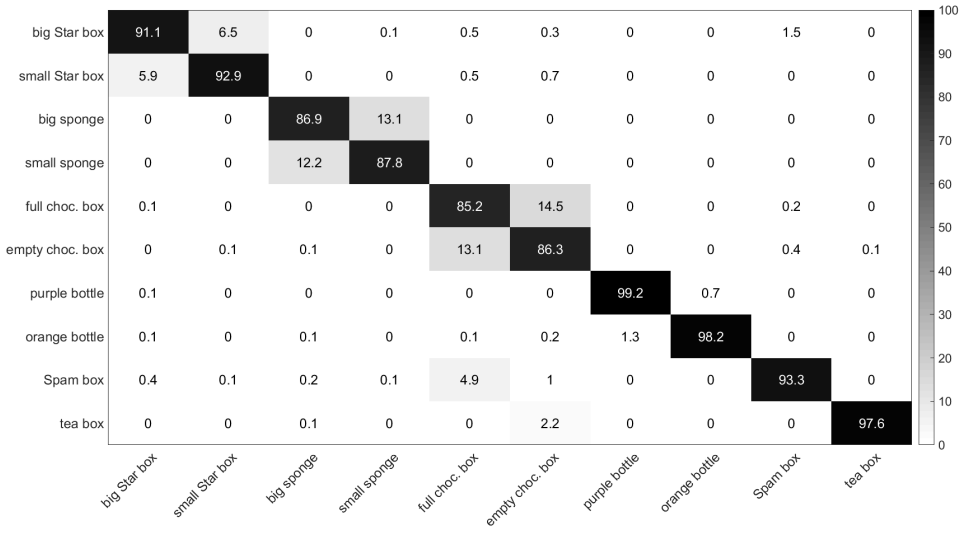


FIGURE 3.15: The confusion matrix of the multimodal classifier trained with the *normalized* features.

3.6 Conclusions

In this work we proposed a method for in-hand object recognition that makes use of a grasp stabilizer and two exploratory behaviours: squeezing and wrapping the fingers around the object. The grasp stabilizer plays two important roles: a) it prevents the object from slipping and facilitates the application of exploratory behaviours, and b) it moves the object to a more stable position in a repeatable way, which makes the learning algorithm more robust to the way in which the robot grasps the object. We demonstrate with a dataset of 30 objects and the iCub humanoid robot that the proposed approach leads to a remarkable recognition accuracy (99.0% $\pm$ 0.6%), with a significant improvement of 29% with respect to the benchmark, in which the grasp stabilizer is not used. This proves that a reliable exploration strategy (e.g. squeezing and re-grasping) is fundamental to acquiring structured sensory data and improve object perception.

Furthermore, we combined our tactile classifier with a previously developed visual classifier, obtaining a higher-level multimodal classifier that resolves the ambiguities of each individual monomodal classifier, improving sensitively the recognition accuracy.

Chapter 4

CONCLUSION

This thesis addressed the problem of robotic tactile manipulation, that is manipulating an object with a robotic hand relying solely on the tactile feedback, without any support of the vision. Indeed, even when vision is available, it may be limited due to poor lighting conditions, or it may be obstructed by the manipulating hand itself. Moreover, some object properties such as softness and texture can be detected only by performing active tactile exploration. Specifically, the topics we focused on are stable object grasping and tactile object recognition.

Stable object grasping is fundamental to prevent objects from slipping and is the base for any kind of manipulation. In ch. 2 we presented our approach for stable object grasping and simultaneous grip strength control. We achieve a hierarchical control by combining techniques from control theory and machine learning. The main novelty of our method is that it decouples the problem of grip strength control and grasp stability, which allows for an independent control of the grip strength while performing a stable grasp. This could be beneficial for several tasks. For example, it lets to squeeze an object while maintaining the grasp stable, or it can be used to perform controlled slip by properly decreasing/increasing the grip strength.

Despite the proved effectiveness of our method, there are still some limitations. The first lies in the assumption on the structure of the robotic hand. Specifically, the high-level controller component, in charge of controlling the object position, assumes that the hand kinematics restricts the fingers to apply forces only along two main (opposed) directions (see Fig. 2.2). In particular, this is true for anthropomorphic hands, in which the index, middle, ring and little fingers apply forces approximately along the same direction and are opposed to the thumb. Extending the method to other kind of hands would lead to sensitively increase the complexity of the high-level controller, whose simplicity is one of the strengths of our method. Fortunately, most of currently available robotic hands match this condition.

The second limitation is related to the transition between the moment in which the object is given to the robot and the completion of the grasp stabilization. Indeed, the pose optimality is achieved only at the end of this phase, while there is no stability guarantee during the transition. In our implementation we limited the problem by applying a smoothing filter causing each joint to move smoothly toward the final hand configuration. Despite this may help, the transition involves the simultaneous movement of several hand joints that could still lead to unstable intermediate hand configurations. Therefore, as a future work it would be interesting to exploit the Gaussian Mixture Model trained with stable grasps also during the transition phase, to properly coordinate the fingers movements and avoid unstable grasps.

The other topic faced in this thesis is tactile object recognition. In ch. 3 we described a novel method for in-hand object recognition that adopts the previously developed grasp controller to stabilize a grasped object. After the stabilization two exploratory behaviours are applied: squeezing to get information about the softness of the object, and wrapping all the fingers around the object to get information about its shape. Using the grasp controller is the strong point of the method, because reaching a stable and repeatable position results in high repeatability of features, which improves the classification accuracy of the final classifier.

However, in order to achieve a good recognition accuracy, the data collection of the training set needed to be carried out in different sessions distant in time. This was done to let the classifier be more robust with respect to the internal "state" of the robotic hand, i.e. the hysteresis level at the fingertips, the friction of the tendons etc. that are generally different according to the specific moment in which the robot is being used. However, the data collection process could be simplified by developing more elaborate features capable of better generalizing over such conditions. These features would also turn to be useful to let the iCub recognize objects with a different hand than the one used for the data collection of the training set.

Finally, it is worth mentioning the main problems encountered while working with the iCub tactile sensors. The first was related to the fact that different areas on the same fingertip generally present different sensitivity. Specifically, the flatter the surface, the higher the sensitivity. Such problem has not been actually solved, but the stability performance of the grasp controller was not considerably affected. Indeed, due to the way it has been designed, the grasp controller automatically compensates for any lack of sensitivity to

eventually reach the target hand configuration. Despite this, when sensitivity is very low at the points of contact, grip strength may be inaccurate. This problem could be tackled by accurately mapping, using machine learning techniques, taxel activations in different areas of the fingertip to the actual force that caused the activations, measured using force sensors. Such method was partially explored but still needs to be improved.

Another problem was represented by a not negligible hysteresis effect. In the experiments described in this thesis such problem has been partially faced according to the specific task carried out. When performing a stable grasp, the hysteresis level at the fingertips was stored right before getting in contact with the object and used to continuously adjust the actual force references sent to the force controllers during the grasp stabilization process. Conversely, for the the tactile object recognition task the best way to tackle the problem was simply to keep changing the order of the objects presented at the robot during the data collection process. This let the classifier be more robust with respect to hysteresis. However, a better approach would be to face the problem at low level by properly modeling the hysteresis effect over time, so that high-level tasks could rely on a better estimate of the actual pressure at the fingertips.

Bibliography

[1] A. Schmitz, U. Pattacini, F. Nori, G. Metta, and G. Sandini, "Design, realization and sensorization of a dextrous hand: the icub design choices,"," in *Proc. IEEE-RAS Int. Conf. on Humanoid Robots*, vol. 186.

[2] G. Vezzani, M. Regoli, U. Pattacini, and L. Natale, "A novel pipeline for bi-manual handover task," *Advanced Robotics*, pp. 1–14, 2017.

[3] G. Pasquale, C. Ciliberto, F. Odone, L. Rosasco, and L. Natale, "Teaching icub to recognize objects using deep convolutional neural networks," in *Machine Learning for Interactive Systems*, 2015, pp. 21–25.

[4] R. Johansson and G. Westling, "Roles of glabrous skin receptors and sensorimotor memory in automatic control of precision grip when lifting rougher or more slippery objects," *Experimental brain research*, vol. 56, no. 3, pp. 550–564, 1984.

[5] A. Streri and J. Féron, "The development of haptic abilities in very young infants: From perception to cognition," *Infant Behavior and Development*, vol. 28, no. 3, pp. 290–304, 2005.

[6] M. Molina and F. Jouen, "Manual cyclical activity as an exploratory tool in neonates," *Infant Behavior and Development*, vol. 27, no. 1, pp. 42–53, 2004.

[7] R. Joseph, "Fetal brain behavior and cognitive development," *Developmental Review*, vol. 20, no. 1, pp. 81–98, 2000.

[8] H. Yousef, M. Boukallel, and K. Althoefer, "Tactile sensing for dexterous in-hand manipulation in robotics—a review," *Sensors and Actuators A: physical*, vol. 167, no. 2, pp. 171–187, 2011.

[9] A. Schneider, J. Sturm, C. Stachniss, M. Reisert, H. Burkhardt, and W. Burgard, "Object identification with tactile sensors using bag-of-features," in *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*. IEEE, 2009, pp. 243–248.

[10] Z. Pezzementi, E. Plaku, C. Reyda, and G. D. Hager, "Tactile-object recognition from appearance information," *IEEE Transactions on Robotics*, vol. 27, no. 3, pp. 473–487, 2011.

[11] G. Metta, G. Sandini, D. Vernon, L. Natale, and F. Nori, "The icub humanoid robot: an open platform for research in embodied cognition," in *Proceedings of the 8th workshop on performance metrics for intelligent systems*. ACM, 2008, pp. 50–56.

[12] G. Metta, L. Natale, F. Nori, G. Sandini, D. Vernon, L. Fadiga, C. Von Hofsten, K. Rosander, M. Lopes, J. Santos-Victor *et al.*, "The icub humanoid robot: An open-systems platform for research in cognitive development," *Neural Networks*, vol. 23, no. 8, pp. 1125–1134, 2010.

[13] L.-W. Tsai, *Robot analysis: the mechanics of serial and parallel manipulators*. John Wiley & Sons, 1999.

[14] A. Sahbani, S. El-Khoury, and P. Bidaud, "An overview of 3d object grasp synthesis algorithms," *Robotics and Autonomous Systems*, vol. 60, no. 3, pp. 326–336, 2012.

[15] D. Prattichizzo and J. C. Trinkle, "Grasping," in *Springer handbook of robotics*. Springer, 2008, pp. 671–700.

[16] J. Bohg, A. Morales, T. Asfour, and D. Kragic, "Data-driven grasp synthesis—a survey," *IEEE Transactions on Robotics*, vol. 30, no. 2, pp. 289–309, 2014.

[17] R. S. Dahiya, G. Metta, M. Valle, and G. Sandini, "Tactile sensing—from humans to humanoids," *Robotics, IEEE Transactions on*, vol. 26, no. 1, pp. 1–20, 2010.

[18] H. Yussof, J. Wada, and M. Ohka, "Grasp synthesis based on tactile sensation in robot manipulation of arbitrary located object," in *2009 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*. IEEE, 2009, pp. 560–565.

[19] H. Dang, J. Weisz, and P. K. Allen, "Blind grasping: Stable robotic grasping using tactile feedback and hand kinematics," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*. IEEE, 2011, pp. 5917–5922.

[20] A. T. Miller and P. K. Allen, "Graspit! a versatile simulator for robotic grasping," *IEEE Robotics & Automation Magazine*, vol. 11, no. 4, pp. 110–122, 2004.

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

[22] J. Felip and A. Morales, "Robust sensor-based grasp primitive for a three-finger robot hand," in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2009, pp. 1811–1816.

[23] M. Ciocarlie, C. Goldfeder, and P. Allen, "Dexterous grasping via eigengrasps: A low-dimensional approach to a high-complexity problem," in *Robotics: Science and Systems Manipulation Workshop-Sensing and Adapting to the Real World*, 2007.

[24] J. Schill, J. Laaksonen, M. Przybylski, V. Kyrki, T. Asfour, and R. Dillmann, "Learning continuous grasp stability for a humanoid robot hand based on tactile sensing," in *2012 4th IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob)*. IEEE, 2012, pp. 1901–1906.

[25] J. Steffen, R. Haschke, and H. Ritter, "Experience-based and tactile-driven dynamic grasp control," in *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*. IEEE, 2007, pp. 2938–2943.

[26] L. Ascari, U. Bertocchi, P. Corradi, C. Laschi, and P. Dario, "Bio-inspired grasp control in a robotic hand with massive sensorial input," *Biological cybernetics*, vol. 100, no. 2, p. 109, 2009.

[27] H. Dang and P. K. Allen, "Grasp adjustment on novel objects using tactile experience from similar local geometry," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2013, pp. 4007–4012.

[28] R. Jakel, S. R. Schmidt-Rohr, Z. Xue, M. Losch, and R. Dillmann, "Learning of probabilistic grasping strategies using programming by demonstration," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*. IEEE, 2010, pp. 873–880.

[29] J. Tegin, S. Ekvall, D. Kragic, J. Wikander, and B. Iliev, "Demonstration-based learning and control for automatic grasping," *Intelligent Service Robotics*, vol. 2, no. 1, pp. 23–30, 2009.

[30] E. L. Sauser, B. D. Argall, G. Metta, and A. G. Billard, "Iterative learning of grasp adaptation through human corrections," *Robotics and Autonomous Systems*, vol. 60, no. 1, pp. 55–71, 2012.

[31] M. Li, Y. Bekiroglu, D. Kragic, and A. Billard, "Learning of grasp adaptation through experience and tactile sensing," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*. Ieee, 2014, pp. 3339–3346.

[32] B. Huang, S. El-Khoury, M. Li, J. J. Bryson, and A. Billard, "Learning a real time grasping strategy," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*. IEEE, 2013, pp. 593–600.

[33] D. Gunji, T. Araki, A. Namiki, A. Ming, and M. Shimojo, "Grasping force control of multi-fingered robot hand based on slip detection using tactile sensor," *Journal of the Robotics Society of Japan*, vol. 25, no. 6, pp. 970–978, 2007.

[34] J. Jalani, N. Mahyuddin, G. Herrmann, and C. Melhuish, "Active robot hand compliance using operational space and integral sliding mode control," in *2013 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*. IEEE, 2013, pp. 1749–1754.

[35] T. Feix, R. Pawlik, H.-B. Schmiedmayer, J. Romero, and D. Kragic, "A comprehensive grasp taxonomy," in *Robotics, Science and Systems: Workshop on Understanding the Human Hand for Advancing Robotic Manipulation*, 2009, pp. 2–3.

[36] V. Mathiowetz, N. Kashman, G. Volland, K. Weber, M. Dowe, S. Rogers *et al.*, "Grip and pinch strength: normative data for adults," *Arch Phys Med Rehabil*, vol. 66, no. 2, pp. 69–74, 1985.

[37] S. Takamuku, G. Gomez, K. Hosoda, and R. Pfeifer, "Haptic discrimination of material properties by a robotic hand," in *Development and Learning, 2007. ICDL 2007. IEEE 6th International Conference on*. IEEE, 2007, pp. 1–6.

[38] M. Johnsson and C. Balkenius, "Recognizing texture and hardness by touch," in *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*. IEEE, 2008, pp. 482–487.

[39] S. J. Lederman and R. L. Klatzky, "Hand movements: A window into haptic object recognition," *Cognitive psychology*, 1987.

[40] J. Hoelscher, J. Peters, and T. Hermans, "Evaluation of tactile feature extraction for interactive object recognition," in *Humanoid Robots (Humanoids), 2015 IEEE-RAS 15th International Conference on*. IEEE, 2015.

[41] D. Xu, G. E. Loeb, and J. A. Fishel, "Tactile identification of objects using bayesian exploration," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*. IEEE, 2013, pp. 3056–3061.

[42] S. Chitta, M. Piccoli, and J. Sturm, "Tactile object class and internal state recognition for mobile manipulation," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*. IEEE, 2010, pp. 2342–2348.

[43] V. Chu, I. McMahon, L. Riano, C. G. McDonald, Q. He, J. M. Perez-Tejada, M. Arrigo, N. Fitter, J. C. Nappo, T. Darrell *et al.*, "Using robotic exploratory procedures to learn the meaning of haptic adjectives," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*. IEEE, 2013, pp. 3048–3055.

[44] M. Kaboli, R. Walker, G. Cheng *et al.*, "In-hand object recognition via texture properties with robotic hands, artificial skin, and novel tactile descriptors," in *Humanoid Robots (Humanoids), 2015 IEEE-RAS 15th International Conference on*. IEEE, 2015, pp. 1155–1160.

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

[46] B. Higy, C. Ciliberto, L. Rosasco, and L. Natale, "Combining sensory modalities and exploratory procedures to improve haptic object recognition in robotics," in *Humanoid Robots (Humanoids), 2016 IEEE-RAS 16th International Conference on*. IEEE, 2016, pp. 117–124.

[47] A. Tacchetti, P. K. Mallapragada, M. Santoro, and L. Rosasco, "Gurls: a least squares library for supervised learning." *Journal of Machine Learning Research*, vol. 14, no. 1, pp. 3201–3205, 2013.

[48] B. Calli, A. Walsman, A. Singh, S. Srinivasa, P. Abbeel, and A. M. Dollar, "Benchmarking in manipulation research: The ycb object and model set and benchmarking protocols," *arXiv preprint arXiv:1502.03143*, 2015.

[49] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[50] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.