Università di Pisa

Facoltà di Ingegneria

*Corso di Laurea Magistrale in Ingegneria Robotica e dell'Automazione*

Tesi di Laurea Specialistica

# Design, development and control of a soft robot for object manipulation in Amazon factory-like environment

**Relatore**:
Prof. Antonio Bicchi

**Controrelatore**:
Prof. Carlo Alberto Avizzano

**Autore**:
Tommaso Pardi

Anno Accademico 2014-2015

**Abstract**

This thesis deals with the problem of automated object handling in a factory-like environment using soft robotics technology. Novel soft robotics actuators that can modulate their own compliance are a recent promising technology capable to safety interact with unstructured environments. In this thesis we investigate a recent control approach for soft robots based on the exploitation of environment constrain (EC) to help the robot perform grasp and manipulation tasks. We designed a *Pick 'n' Place* manipulator composed of Variable Stiffness Actuators (VSA) and a Pisa/IIT SoftHnad, an under-actuated anthropomorphic hand with 19 DOFs but only one motor, characterized by outstanding adaptation capabilities. Moreover we embedded in the system a three-dimensional depth sensor to reconstruct the geometry of the actual scene. In the first part of the thesis, we experimentally analyzed how the Pisa/IIT SoftHand behaves in typical application cases. Using a manual handle device to approach objects in a human-like way, information was gathered in a grasp database, in the next step we elaborated manipulation strategies based on the previous results for a set of objects of various dimension. A careful observation of the different strategies let us summarize them in subgroups that depend on objects properties. Finally we tested the devised strategies in test environment designed to match the standard shelf adopted by Amazon in its storage facilities. This case study served as preliminary test before our participation to the *Amazon Picking Challenge*, a competition among international universities and research centers that was held in Seattle during the last ICRA conference.

**Sommario**

In questa tesi si è trattato il problema della manipolazione automatizzata di oggetti all'interno dell'ambito industriale attraverso l'utilizzo della soft robotica. La recenti attuatori *soft* possono modulare la propria *compliance* e rappresentano una promessa, in campo tecnologico, di alte capacità operazionali mantentendo, allo stesso tempo, alti standard di sicurezza anche in ambienti non strutturati. Nella tesi abbiamo analizzato uno degli approcci al controllo basato su soft robots che fonda le proprie radici sullo sfruttamento dei vincoli ambientali (EC) per ottenere prestazioni di presa e manipolazione migliori.

Abbiamo progettato un manipolatore *Pick 'n' Place* composto da Variable Stiffness Actuators (VSA) e con la Pisa/IIT SoftHand, una mano antropomorfa sottoattuata con 19 DOF ma con un solo motore, caratterizzata da una forte capacità di adattamento all'ambiente e all'oggetto su cui si opera. Inoltre abbiamo installato un sensore di pronfondita 3D per ricostruire la geometria della scena attuale.

Nella prima parte della tesi abbiamo analizzato sperimentalmente in che modo la Pisa/IIT Softhand si comporta in tipiche applicazioni di utilizzo. Attraverso uno strumento manuale, handle, abbiamo cercato delle strategie *human-like* di approccio a diversi oggetti, le informazioni sono state raccolte all'interno di un DataBase, nel passo successivo abbiamo testato le strategie di presa su un insieme di strumenti con varie dimesioni. Una attenta osservazione delle strategie trovate ci ha permesso di applicare una riduzione ulteriore di questi gruppi basandoci sulle caratteristiche degli oggetti.

Nell'ultima fase abbiamo testato le strategie in un ambiente di studio progettato per ricalcare le caratteristiche di uno scaffale standard usato da Amazon nei propri centri di smistamento. Questo caso di studio è servito come elemento preliminare alla partecipazione della *Amazon Picking Challenge*, una competizione tra universita internazionali e centri di ricerca che si è tenuta a Seattle durante la conferenza ICRA.

# Contents

# Chapter 1

# Introduction

Since born moment every humans interact with environment and this simple event became so common that most of us does not ask himself how it happens in all entire life. The reasons behind this lack of interest is not to search in a human casualness and, of course, not in poor interest of the subject, so why? Like a lot of questions the response is basically easy and in straightforward view: simply we don't need to know consciously.

In fact, in the early years of a human being life the brain is barrage from new information comes from outside and, in the meanwhile, he has a body to manage and learn to use properly.

We use eyes to get a three-dimensional image of objects and people inside an environment, we use hands and muscles to move toward and touch things, further our skin allows us to get informations about the kind of material and his solidity; these are only few examples about how, and what kind of informations, our brain gather.

There is not an out-and-out way of education but it is different from everyone skills and environment, nevertheless in all cases a lot of several exogenous inputs excite the five senses and, from these, our body decides how to understand it and a later of time how to reuse it or integrate them with others previous events.

The process is hard and employed about the first three years, the climax is at beginning and lower in time, to have an almost complete form, rest of gestures or moves are learn during the whole life time of subject and depend on his abilities and talents.

After this period we are able to interact with objects and persons, we are able to manage our strength to be assured on the attaching and we are able to perform a lot of other tasks joining what we see with what we know to exploit a specific situation, representing a dynamic way of think and act; of course this is not a neuro-science document and we don't want go down in details about that, but it is very interesting understand how this long process happens and how his conclusions can represent a baseline, or in some case a turn around, for different branches of science which try to imitate what human being does or, even better, try to join this knowledge with other field of interest, in this way we want to combine two or more different worlds in one.

## 1.1 Robots for physical interaction

Robot world comes from an English translation of *robotnik*, Czech name, which has a approximate meaning of "forced labor" and at beginning of robot structures creation they was used only like hard work assistant or substitute.

The first mechanical structure of this type was design by George Devol in 1950s and his names was *Unimate*, Fig. 1.1, the only job of *Unimate* was pick a die casting from an assembly line and welding it on auto bodies, these task was very dangerous for human been because of toxic fumes present in scene.

Such kind of robot is used mainly in industrial automation work environment and it experienced a lot of improvement from this early stage up to structure which perform more than only task at same time and which follow rules to guarantee human safety too.

Another branch of robotic run parallel to industrial automation and it is represented by robot which imitate or try to perform typical human task, like walk, manipulation of object inside general environment or interact with persons and objects but keeping safety for users in mind.

This second class of structures is very interesting and, in last few years, it went through an intense progress which permitted to perform most of human gestures, like walk, grasp, manipulate objects, watch and etc., developing new technology. But it is not enough, thanks to research engineers developed new mechanical and electrical device which are used to create hi-tech prosthesis, air drones so on; all of this is possible creating a relationship from different branches of knowledge, which act together to the same goal.

Investigating these new technology to understand better why they are so essential; mathematic analytics of dynamic movements and static positions allow us to know how a general mechanical structure can move, usually we talk about joints, or series of joints, which represent the moving parts of robot, Fig.1.2, and due to control theory robot is able to reach a specific goal positions and attitude inside a feasible set.

But theory model is not enough to design a robot, we need a mechanical structure which is subject to desired task specifications, in a general way we divide in two pieces a robot: the framework, which is the static part, and motors, which are employ to give movements.

With technology improving the motor structure is changed moving his basics from mechanic to electric mode of operation but is not all, common motors are rigid so when an opposing torque is applied a proportional command is generated to (ideally) neutralize it holding position, moreover rigidity of these kind of devices can be dangerous for users, but recently a new vision was proposed.

Since early years of 1990s the design focus was moved to robot witch can interact with humans keeping safety, this approach light up the Variable Impedance Actuators (VIA).

First VIA was developed in 1991 by Laurin-Kovitz and his main property is possibility to change his own position equilibrium when an external torque or force is apply, this capability enter in the design process where we find damper factor sources and/or springs; when external

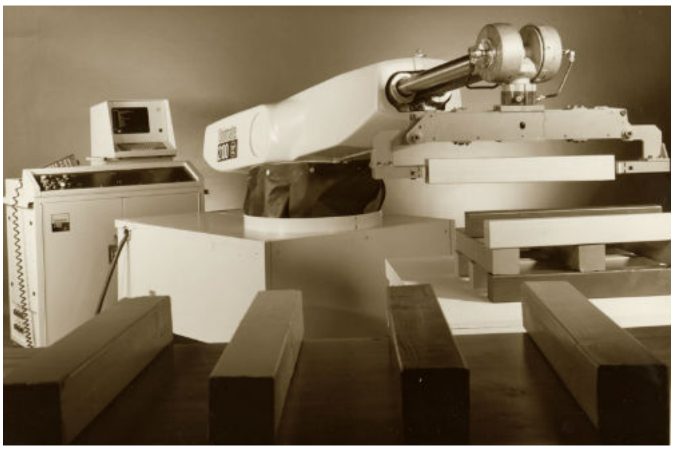


Figure 1.1: Unimate, industrial robot created by General Motors in 1961.

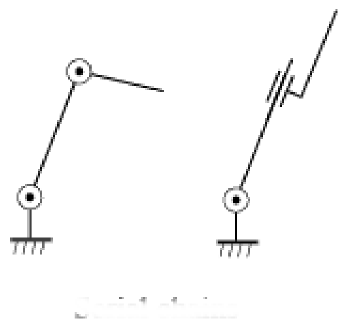


Figure 1.2: Serial joints chain.

force/torque stop motors reach equilibrium position mechanically.

Among VIA models we find Variable Stiffness Actuators (VSA), first VSA prototype is design to Bicchi et al. in 2003 and it has a mass - spring - damper dynamic. Researcher study several ways to get this properties and it bring to life different mechanical approaches, example how to configure springs on shaft.

Another important point of view regarding robot is the outside vision, not only figurative but also in an optical way of think, thanks lasers and depth sensors we are able to have a telemetry of space surround us like eyes in humans does.

Elaborating this kind of informations through complex algorithm we can decide how move robot to avoid obstacles or manipulating objects, last element of a robot is software which represent his brain and allows device to take decisions autonomously, in this way a request task can be accomplish in specific situation through general purpose, it blend together all informations come from sensors and use them to give necessary control command.

Depends on what is our task we are able to classify different field of interest, in Fig.1.3:

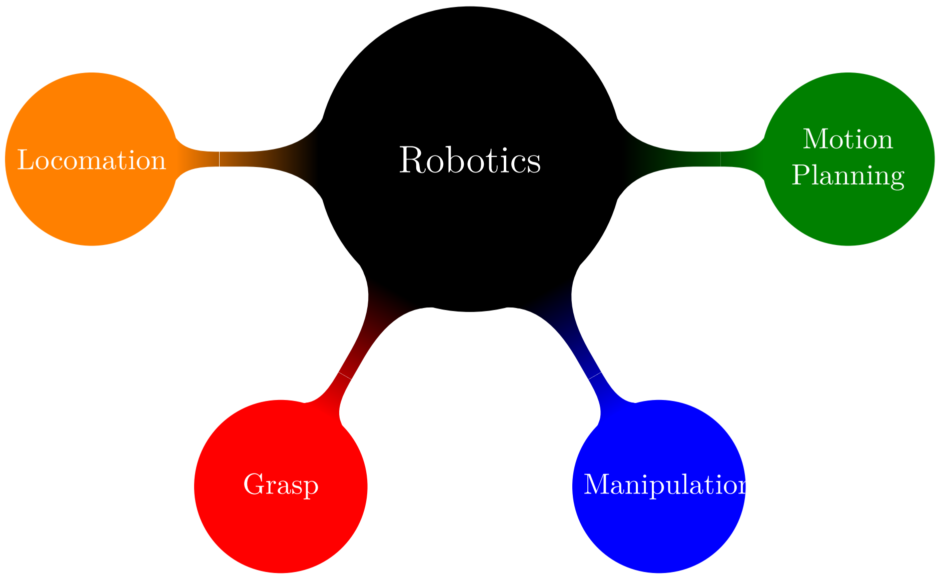



Figure 1.3: Robotics branch.

Now we will focus our attention about *manipulation* which is the main part of this degree thesis.

## 1.2 Manipulation

At the beginning we define what it means:

> *objects manipulation* - Its the action of move, grasp, release and use objects which have a specific pose and attitude with the goal of pull in, pull out or change positions of objects from a platform interacting with that by the robot mechanical structure.

Definition is very general and cover a lot of tasks and possibility but reading it with attention two actors show up clearly: manipulator structure and object manipulated; but there is another player, maybe a little bit more hide, the environment in fact both object and end-effector has to relate with it in some way.

The tool interacting with outside components is call end-effectors, or gripper, and his form depends on task peculiarity, we can find an heterogeneous kind of grippers from the traditional clamps to high sensored three-finger grippers or even out-and-out hands with five-fingers, fully sensored or strongly under actuated.

There is a entire world of different end-effector which put first a specific component in his functionality, so only a meticulous study of the problem show up what is the best choice.

Object analysis is more complex because of admissible range, every dimension or structure is possible and it is the gripper which have to adapt to it.

Now talk about environment, from this side we can identify two different approaches:

- Environment can be part of the problem
- Environment can be part of the solution

### 1.2.1 Environment as problem

Under assumption of seeing environment like a problem it give us movement constrains so a set of positions and attitudes which are blend together creating the *C-Space* set, Configuration space, it represent all feasible configuration for robot which does not touch anything.

Until some years ago this school of thinking was the unique because robots used was very rigid and also a little impact could be disastrous, a depth study of problem produce different ways to exploit problems through algorithms and control rules relevant for general robot structures.

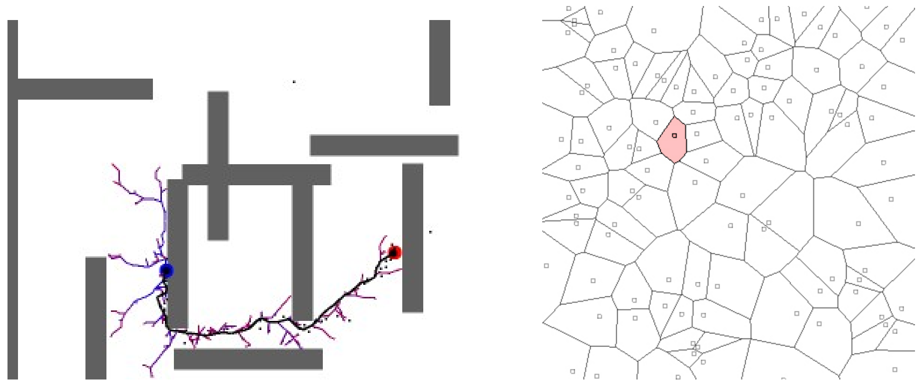
Manipulation is composed by consecutive steps and in a example task we look for an initial grasping stage, a collision avoidance stage and a final releasing stage.

All stages are fasten but collision avoidance is the branch more strongly connected with environment than other, when we understand how take a specific object from a place to maximize percentage of grasp also we want to know how we can reach this configuration without collide with other object in the scene or hit end-effector in one of possible environmental constrained directions.

The problem is very huge and deserve an examination in depth of the topic, it's enough ask himself: what happens when end-effector hit another objects or, worst, a wall? In the first event the collided object can moves target and we need a further study on the new position to understand how approach it, second hypothesis can be catastrophic, depend on robustness of *e-e* structure blow can break the gripper or break the wall instead.

How I previously mentioned a lot of different algorithms was proposed and studied which resolve, in varied ways, the same problem, the main division is based on idea behind how to choose feasible set points, some of that use statistic ideas based on Monte-Carlo method, examples are RRT [LaV98] and RRT*, the other are old-fashioned way and try resolve problem in a deterministic way, an example is Voronoi[Aur91], both of them have *pro* and *contro*, so the criteria for choosing depends on problem properties and specifications.

Benefits of random based algorithms are greater when we use them on unknown world in fact points are random generated and only after that we check feasibility, another important advantage is CPU consumption because this kind of algorithm get a solution after few iterations, on the contrary optimality need a lot of iterations (so time). The work order is simple they get randomize points inside set's boundary and collect them in a tree structure, like KD-tree, which take trace of all points position and permits to find the minimum distance path to reach a chosen point.



(a) (b)

Figure 1.4: (a) - RRT example, (b) - Voronoi example.

Deterministic algorithm family use several kind of strategies for reach a specified position, most of them use point of interests in the action range and define a structure to collect them, after this fist stage it analyze all points to get a optimal solution depends on points request, other does not use structure based on points but focus their attentions on distances from hyper-planes in the moving field.

### 1.2.2 Environment as solution

How we discuss previously, humans been does not separate in different problems manipulation task consciously but use all informations comes from senses to accomplish it. Humans make a step forward exploiting the compliance of their body to engage in functional interactions with necessary parts of the environment.

This apparently simply way of think is growing only in last years and it open doors to a whole new bunch of options which try to solve the same problem than before from a different point of view.

So Environment Constrains (EC) help us to accomplish our goal: grasp or manipulate; but technology have to make a step forward too.

How can we use a rigid structure and end effector, which does not have compliance, to perform human-like operations? We can not.

We had already talk about VSA which represent an option to make compliance structure but we need an improvement of *e-e* in the same direction. The new kind of manipulator use soft mechanical design which tolerate of the gripper without damage, Nowadays only few research centers design and produce this kind of devices and the main examples are Pisa/IIT SoftHand from Research Center E.Piaggio or RBO hand from TUB.

TUB researchers proposed a solution based on EC, in a preliminary stage they search possible EC in a scene, like wall, floor or ceiling, with three-dimensional camera sensor, in a second stage plan their grasp through different sets using the same EC to set limitation on moving, in this way they eliminate the existing separation from perception, planning and control creating a all include process.

They proposed a blending together all mechanical parts and sensors to exploit grasp in a human-like way.

Obviously grasp rate of success depends on geometry of objects and actual real world EC but they demonstrated a high rate of success after in a real world experiments even with an only three-dimensional image of scene.

Recently a new European project, Soft Manipulation ($\Sigma\Omega MA$), are studying and designing hands, planning algorithms and perceptions around the exploiting of EC concept with goal of improve grasping performance and, in the meanwhile, keeping in mind human safety and robust of mechanical parts.

# Chapter 2

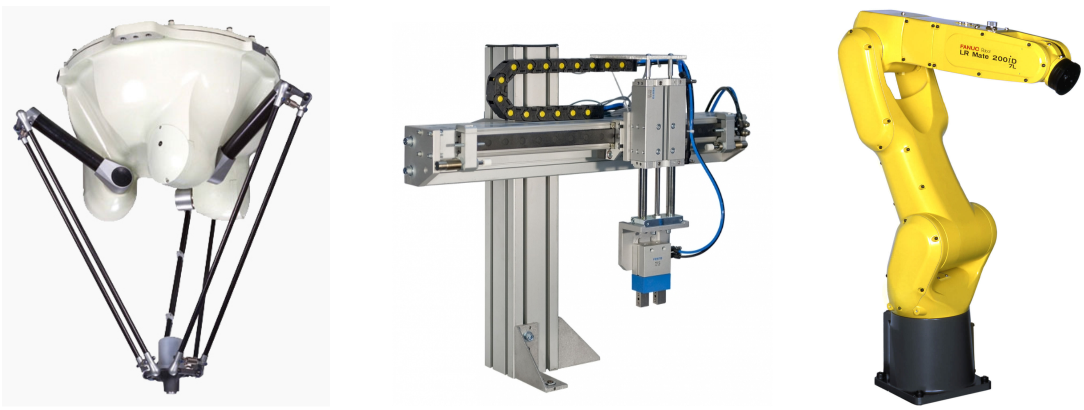
# State of Art

## 2.1 Robotics Structures

### 2.1.1 Pick 'N Place

Pick and place robots, or more precisely surface mount technology (SMT) component placement systems are robotic machine which speeds up the process of picking part up and placing them in new locations, increasing production rates.

The most common purposes of these kind of robots are moving little objects, like resistors, capacitors or integrated circuits, or very heavy components in a different position, for example on a board or between two different conveyor belts.



(a) (b) (c)

Figure 2.1: (a) - Delta robot by Kawasaki, (b) - 2 axis robot, (c) - Anthropomorphic robot by FANUC.

Pick 'n place robots have not an only preset shape but it depend to what structure have to do, we usually can see *Delta robot*, Fig.2.1(a), which are a type of parallel robot capable of perform pick and place operations at very high speed thanks to its structure, often these robots have not a single gripper but they use end-effectors with several tongs to speed up process.

Another big part of pick 'n place sector use robots composed by prismatic e rotating parts, they have motors mounted orthogonally one to another to move itself in essential directions and use rotate motion to permit orientation sets, in Fig.2.1(b) we see a simple example with only *two work axis* instead in Fig.2.1(c) we see an *anthropomorphic arm* which represent a typical complex robot composed by only rotate motors.

Over or between ’90s and 2000s Pentel and NEC company propose a new kind of P&P robots that begin to be always more frequently, it is called *Selective Compliance Assembly Robot Arm*, *SCARA*, which use an actuator chain of rotate motors for the 2-D operations and a further degree of freedom responsible of lift the arm.

This four DOF structure is faster and cleaner than comparable Cartesian robot systems moreover usually it is easy to mount but, on the other hand, it is more expensive than Cartesian opponents and request a inverse kinematic software to interpolate positions.

## 2.2 Grippers

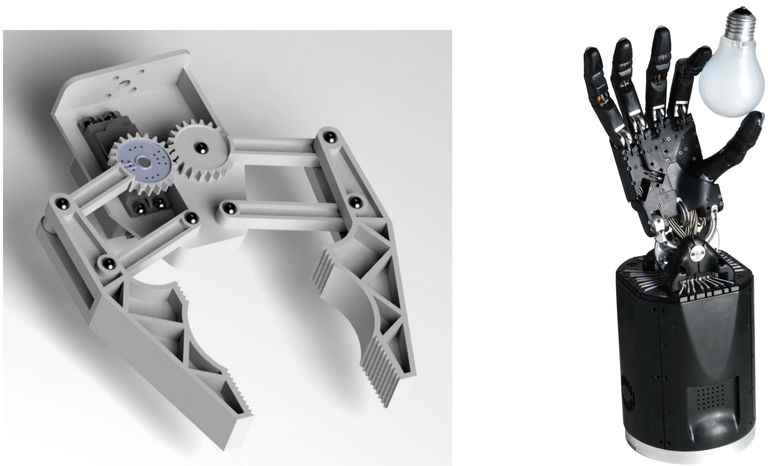
Interact with objects or environments require a tool which is not unique, in fact a lot of factors enter in choosing process: object dimensions, shape, weight or environment dimensions and constrain but outside structural considerations are not the only things, at the same way robustness, safety and cost of the gripper are important too.

All these aspects are very important but we can trade some of them off to maximize a special side characteristic which is essential for a particular operation; under this point of view we find several kind of gripper.

Plier is the simplest end-effector which is a tools composed by two metal small plate used to grasp objects, its simply go to the detriment of grasping capable, for example to pick an object up *e-e* should be rotate in a specific assets moreover it need further sensors add on it to avoid problem during grasping.

One of most popular grippers is *Shadow Dexterous Hand*, in Fig.2.2(b), which imitate human hand to improve grasp properties using tendons and sensors on its to perform absolute position and force torque on each finger, all the twenty DOF on the hand are fully actuated.

On the other hand this kind of grippers are very complex, we need inverse kinematic algorithm to reach a defined position and torque control to avoid problem attaching object, further issues raise when we want command all twenty actuators require to move the hand. These problems often are masked by company which gives us a way to communicate with



(a) (b)

Figure 2.2: (a) - Simple gripper, (b) - Shadow Dexterous Hand by Shadow Robot Company Ltd.

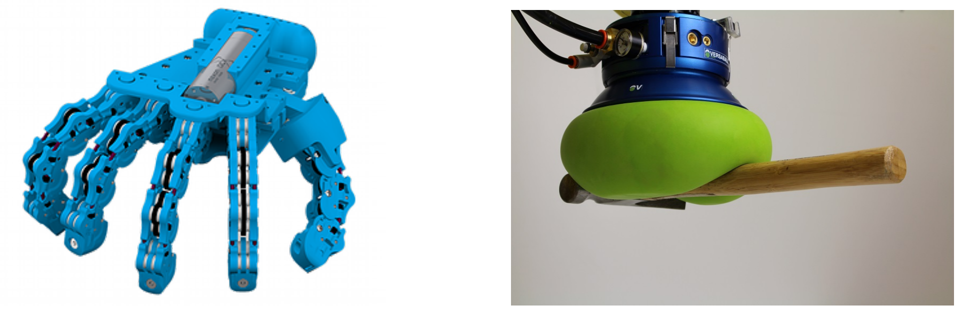
gripper without carry about electrical or mechanical troubles.

Other grippers try to avoid these problems take advantage of different shapes and technology improvement, an example far away from previously is shown in Fig.2.3(b), this device works with vacuum technology which suck in air creating vacuum-seal between end-effector and object surface, when air is release the grasp object fall down. Design of *Versaball* is extremely different from human hand but it is designed to get the best approach on manipulate object in fact it can warp its shape to seal on object surface.

In Fig.2.3(a) is shown *Pisa/IIT SoftHand* design by Center E. Piaggio and IIT (Istituto Italiano Tecnologia) and it represent an U-turn of common approach, researcher made a deep study on human gestures and they discovered that we can divide in path, call synergies, human grasp and every approach or grasp is a combination of these synergies.

On this line of sight Pisa/IIT researcher design a hand based on first synergy which cover up to 90 % of human grasp, moreover they used a tendons system which permit to keep use of it simple; thanks to its properties we can control hand closure with only one motor and when gripper run on an object or an environmental constrain adapt itself echoing the closure on leftover fingers.

Controlling all end-effector movement with an only motor let this hand be perfect for prosthetic art.



(a) (b)

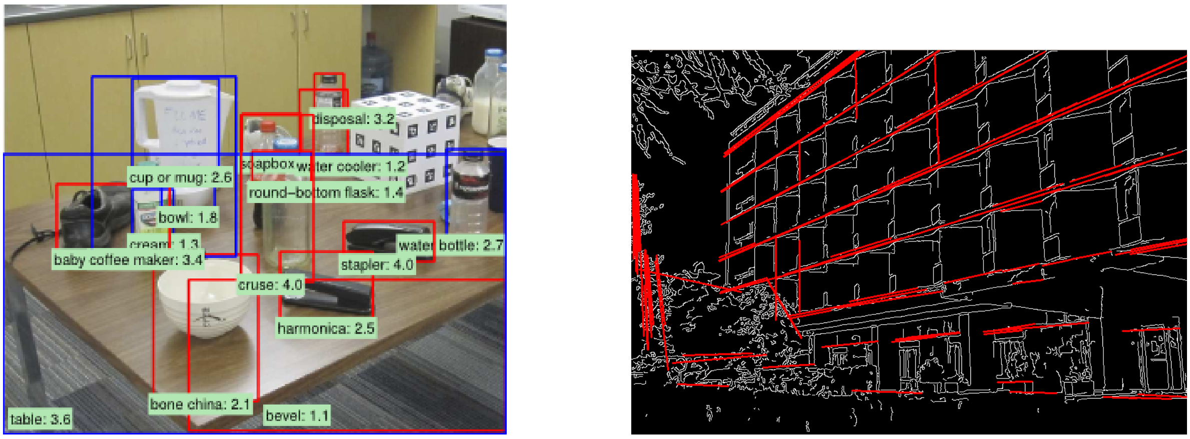
Figure 2.3: (a) - Pisa/IIT Soft Hand by Pisa/IIT , (b) - Versaball by Empire Robotics.

## 2.3 Perception

In literature we can find a lot of object recognition methods, part of them based their action only on 2D image and use a set of information like: dimension, shape, color, gradient and etc. to identify a specific item in a scene, usually they implement algorithms which allow to recognize important part the environment, like corner and edge, by transformation, like Hough one.

These key points inside the image are saved and the method researches them on next image, in this way we can identify dynamic and static element to improve filtering depend on what are we searching.

But recently new technologies improved substantially recognition algorithms, in fact new camera can sense a further degree of freedom giving information about depth of image, these 3D images are called point clouds because they are composed with a set of three-dimensional points and, often information about color of each point.

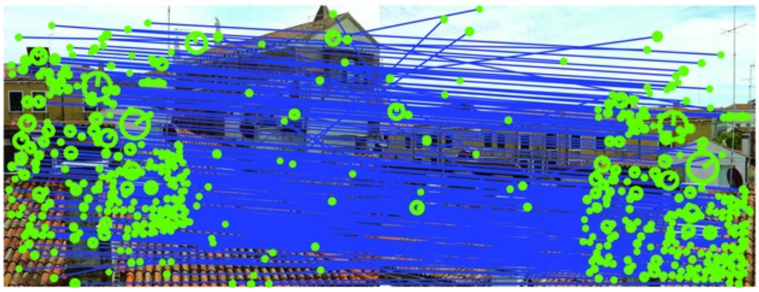



(a) (b)

Figure 2.4: (a) - 2D recognition algorithm example, (b) - Hough transformation with OpenCV.

At the same time new algorithms are proposed to exploit this new feature, one ot these algorithms is based on Scale-invariant feature transform (SIFT) which takes key points inside an 3d image that characterize a specific object.

Properties extracted by objects are collected in a database which is used to identify the same object inside a different scene.



(a)

(b)

Figure 2.5: (a) - SIFT characteristics of an image, (b) - Point cloud example.

Another important algorithm which takes advantage of 3D feature is Iterative Closest Points (ICT) [Bes92], it is based on calculation of distance from two point clouds, after clustering process of the image we test every cluster with all point clouds stored in database, so it represent a *force brute* method with an huge computation time but with a good rate of recognition.

For reduce calculation time usually, before of ICT algorithm, a strong filter on possible cluster is made studying information like shape or dimensions.

## 2.4 Control strategies for Soft Robotics

Many recent advances in robotic grasping and manipulation can be explained by a simple insight: contact with the environment can improve performance.

Under-actuated, soft hands benefit from the interactions that occur naturally between hand, object, and environment.

Furthermore, the robustness of grasping can be increased through the use of contact with support surfaces.

Humans grasp objects employ the environmental contact, especially when grasp task is difficult, Fig.2.7(a)-2.7(c), nevertheless the most of motion planning algorithms reject the exploitation of natural constraint that are present in the scene, on the contrary constraints are seen only as obstacles.

A novel theory overturn the problem trying to exploit environmental constraint to grasp an object.

Environmental constraint (EC) [Epp15] as a feature of the environment that enables replacing aspects of control and/or perception with interaction between hand and environment.

To plan the exploitation of EC, we must eliminate the existing separation between perception, planning, and control. Instead, we tightly integrate perception and action by realizing each to satisfy the others requirements and to account for its limitations [All15], Fig.2.6.

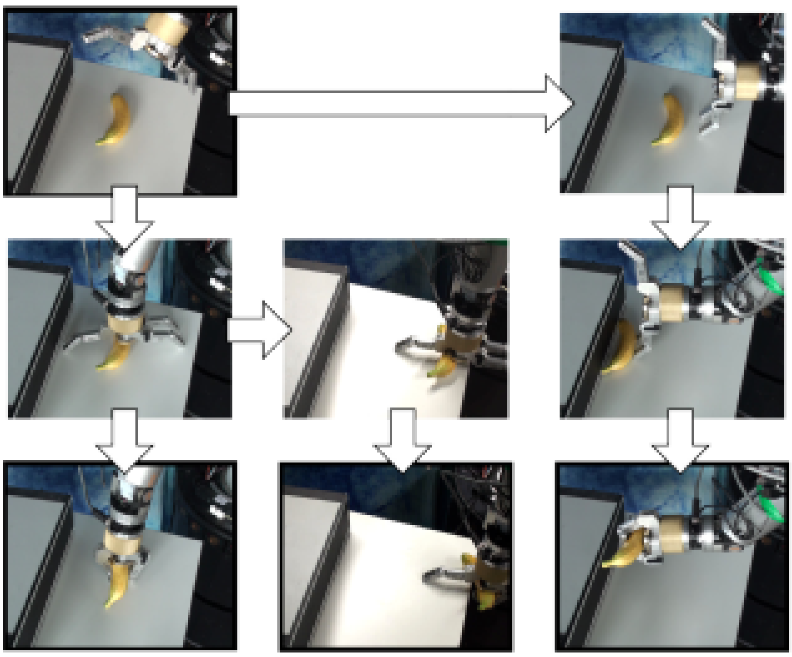
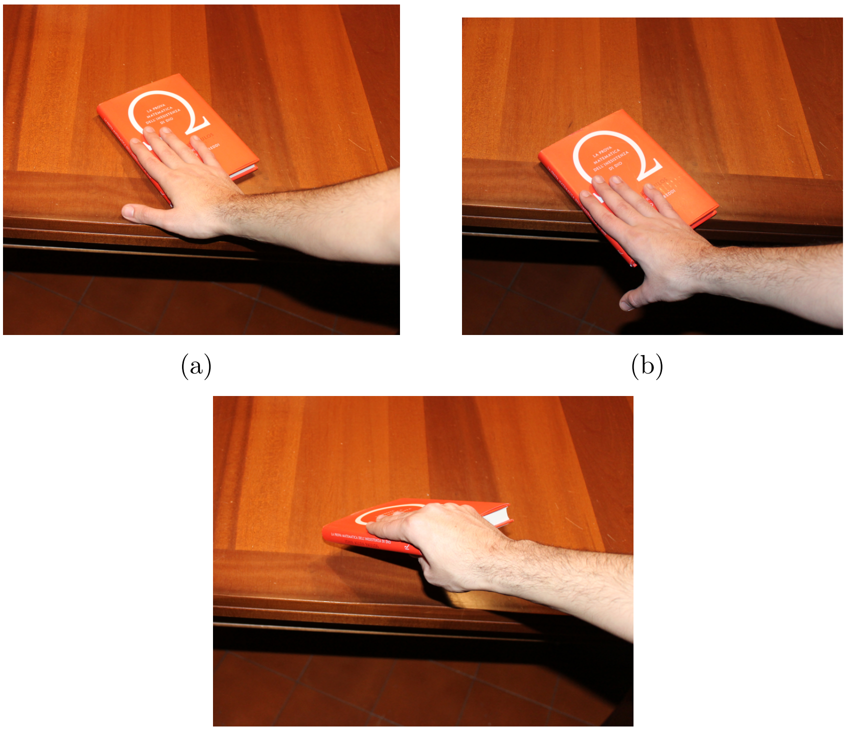


Figure 2.6: Exploit EC with a robot.



(a)

(b)

(c)

Figure 2.7: Human takes a book from a plane.

# Chapter 3

# Problem Definition and Approach

How we already said one of most important problem regard human and robotic interacting, it is a long history and, in the past, scientist propose a lot of different response to solve it.

In first part of time we try to avoid human to interact with robot, in this way human safety is guaranteed and to do it usually we mount robot in particular area surrounded by cages and connected to security system to prevent eventually problems or unexpected breakdown. This way is common still now and, in some part, undeniable, but it fully denial human and robot interaction.

Figure 3.1: Industrial caged robot.

We find a typical example of robot caged in Fig.3.1, where users stay beyond the trespassing area.

But industry is not the only one fields of interest about robots, keeping in mind domestic devices, like cleaning robot or mower, or humanoid robots which are the representation of a blooming sector for research in last years.

In despite of that, in the matter of opportunities, human could strongly improve robot functionality interacting with its, so we can think a way to let coexist both together without danger.

The problem cover several point of view and different approach are available, a well pose questions could be: why is it so dangerous?

Its simple, anthropomorphic robot arms, like in figure, are stiff and they perform their movements fast and precise but, at same time, they stay rigid during all the process. So when a body like human is in a robot trajectory, users get a very strong blow, some time fatal which have to be avoid.

How we explain before the first response is use variable stiffness motors which permit movements more safety than before, but a new troubles arise: how we can control it? or how we can use these new feature to performs better approach or grasp?.

Every news brings new questions and it needs time to stabilize itself; between the huge set of ideas proposed some of that are very interesting and need further investigation about that.

Environment becomes a fellow and no more an enemy, an inclusive point of view is favorite to an opposing way of think, but we need to create different kind of end-effectors too, which can manipulate inside an environment and, at same time, external input can warp the shape without dangerous for gripper or obstacle itself.

In last years a lot of structures with this kind of characteristics are designed and most people studied the best way of use it and test its limits but we only start to scratch the surface of topic.

In order to make step ahead in this field we have to design structure with this specific properties and use them in all possible situation even in typical industrial environment, at the moment only ABB propose an gripper with soft properties which handle fragile things but typical robots are composed by classic rigid actuators.

## 3.1 The Amazon Picking Challenge

Around the world a lot of challenges are crated every years that because they encourage researchers to find out unexplored solution to known or unknown problems, extending actual results while they focus their attention on a specific problem. Under this spirit private and public companies sponsored different kind of challenges with several goal and prize.

This year an important factory, Amazon Inc., decided to announce the first challenge based on Pick and Place task called Amazon Picking Challenge (APC).

Although object manipulation represent a critical issue for factory automation still now it present a lot of several problems which are unsolved so high tech companies want focus research world on this topic.

amazon Picking Challenge
ICRA 2015, Seattle WA

Figure 3.2: Amazon Picking Challenge Logo.

APC took place in Seattle (WA) during 2015 ICRA Conference and it hosted a lot of competitors from all over the world with several kind of approaches to the same problem, we can organize in four simple steps the proposal challenge:

- Interpret a JSON file at runtime
- Recognize request object inside a specific bin
- Pick the Object

- Place the Object in a external box

Robot operations had to be fully automated and it can't interact with an external human user in anyway during all process, the JSON file contained information about all request objects and robot should execute its task until all stuff is taken, or when the team leader decided to stop.

At the beginning of trial a judge pose all elements inside the shelf in a preset position and orientation and, when challenge end, he assign points depending on object taken and their bin position or drop it when a rules is violated, moreover some things bring extra points because of their particular complex shape, Tab. 3.1.

Challenge environment is a classic Amazon shelf, Fig.3.3 so robot have to move in a wide range up to 1.65 m high, 1.0 m width and 0.8 m depth.

Shelf has a grid shape which splits it in twelve bins, with different dimensions, the entire environment; we reported in Fig.3.4 size for each bin.

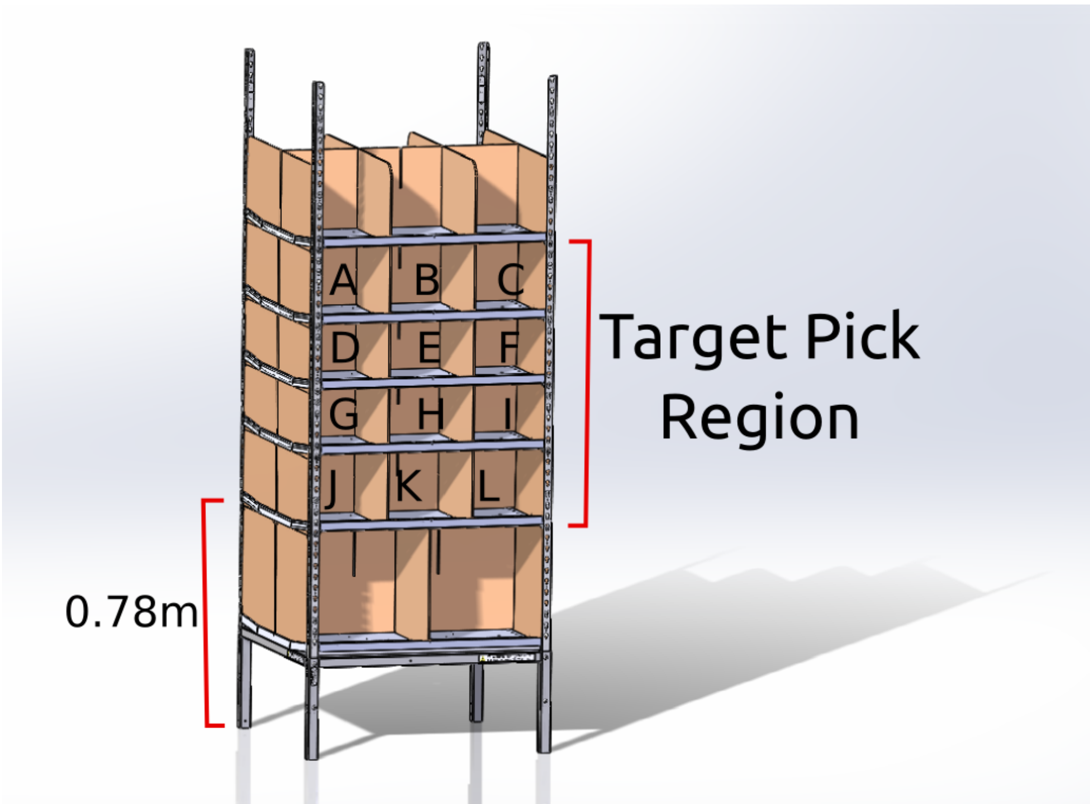



Figure 3.3: Amazon shelf CAD.

The set of potential items that will be stocked inside the bins is shown in the figure 3.5. The graphic below shows the proper form (including things like packaging). The contest shelf may contain all the announced items, or a partial subset of them. All items will be sized and located such that they could be picked by a person of average height (170 cm) with one hand, moreover all items the teams are required to pick from the system will be a subset of this set.

A bin could contain one or more objects but no occlusions are permitted, moreover when an item is taken we get a number of points depends on how much objects are inside.

- **Single-item bins**: At least two bins will only contain one item (single-item bins). Both these items will be picking targets.

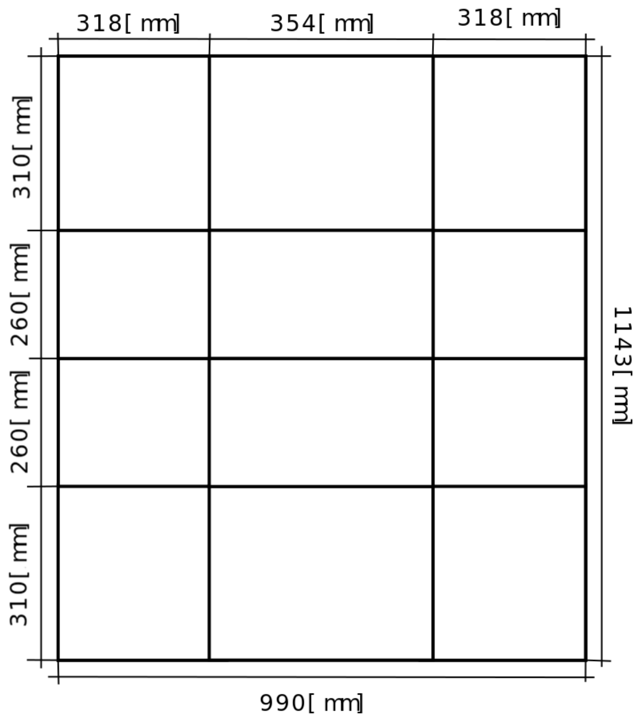



Figure 3.4: Bin dimensions.

- **Double-item bins**: At least two bins will contain only two items (double-item bins). One item from each of these bins will be a picking target.
- **Multi-item bins**: At least two bins will contain three or more items (multi-item bins). One item from each of these bins will be a picking target.

Every teams have two attempts each 20 minutes long, when time elapse, or team leader verbally declare the run complete, robot stop and the judge counts points.

The shelf is placed in a fixed known stationary position prior to the start of the attempt and the team allowed to perform any calibration and alignment steps prior to the start to their timed attempt. An overhead view of the workcell layout can be seen in the diagram Fig.3.6. The team is allowed to place their robot anywhere within the predefined robot workcell, but must have a starting gap of at least 10 cm from the shelf. The robot should be kept within the 2 meter X 2 meter square robot workcell. Each robot must have an emergency-stop button that halts to motion of the robot.

| Action | Points |
|---|---|
| Moving a target item from a multi-item shelf bin into the order bin | +20 |
| Moving a target item from a double-item shelf bin into the order bin | +15 |
| Moving a target item from a single-item shelf bin into the order bin | +10 |
| Target Object Bonus | $+(0 \div 3)$ |
| Moving a non-target item out of a shelf bin (not replacing it in the same bin) | -12 |
| Damaging any item or packaging | -5 |
| Dropping a target item from a height above 0.3 meters | -3 |

Table 3.1: Scoring table.


Figure 3.5: Set of available objects.

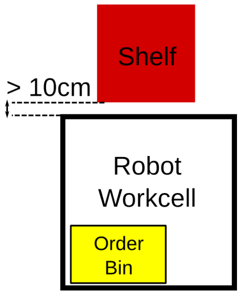



Figure 3.6: Workcell layout.

## 3.2 Solution approach

Keeping in mind previously discussed problems, about human-robot interaction and performing classic task in industry, it suggest that one of most interesting solution is represented by Soft Robotics but at the same time it pose open issues about control and planning.

We investigate about known problems using an approach based on exploiting of environmental constrains, thanks to a structure composed by VSA and Pisa/IIT SoftHand we can perform stiffness modulation and an end-effector with robust and simple properties.

Our investigations are posed in a study frame which is composed by two main parts:

- Perception
- Robot with VSA and Pisa/IIT SoftHand

Both parts are essential when robot has to perform a manipulation task, we should know where the object is and how it is oriented and, only after that, we take the decision about the best grasp strategies for a specific item.

Perception block is composed by a 3D camera which gives us a point cloud of the scene, in a reference tern, after we catch the three-dimensional image from device, a recognition process start to find out position and orientation of the objects. A set of different algorithms implement the recognition process and they sharp solution iterative until an admissible comparison index is reached, moreover algorithm use a database of point clouds which is saved off-line with an *a priori* calibration.

As all sensors camera takes measurements with an error which affect a precise recognition of position, we can consider this problem like a uncertainty on the read value.

We can solve last observation with a motion planner enhanced, checking deflection on VSA motors we can understand when an object is reached, moreover we can eliminate most of determinate movements reducing position mistakes.

After that we get information about object position we would like reach it and grasp it, first part of the problem regard motion planning and second take in account a lot of characteristics, object dimensions and position but also gripper capacity.

We investigated how human takes a set of objects through empirical experiments, we use the set of object in Fig.3.5 because they covered a lot of different dimensions, shapes and stiffness; during this part we used a device call Handle to grasp the selected objects and we gathered all information.

Handle permit to wear Pisa/IIT SoftHand and use it directly, a specific configuration of SoftHand and Handle is used also in prosthetic solution which use electromyographic sensors to get commands from user.

After that we finish experiments we passed to next stage, in the second step we analyzed all information and categorize them in groups; each group represent a way to grasp an object through Pisa/IIT SoftHand with given structure constrains.

We found relation about dimension and shape with specific grasp groups and we summarized original grasp strategies in a more general subset which include the other.

Other big part of project is motion planning, there are a lot of possible solutions about it and we decided to develop a custom code for our goals; grasp strategies blocks produce a series of vector state which contain configuration of each motor and closure rate of Pisa/IIT SoftHand, when a the selected grasp routine produce an enough number of states a trajectory generator start, the block execute requested state on arrival order and imposing the same final position time, we slow down motors which has to travel long distance, after that actual referents of actuators keep steady until next execute from trajectory generator.

When a referent position out from trajectory generator block a PID controller control error to zero, we want maintain stiffness modulation properties during all process so we change integral coefficients in specific situations where deflection is more important than error position.

Last stage of thesis regard implementation of the strategies and further experiments to validate grasp groups find out, moreover our robot competed in APC performing pick 'n place operations on a constrained environment and using a set of object randomize distributed over the shelf.

# Chapter 4

# Hardware

## 4.1 Structure Definition

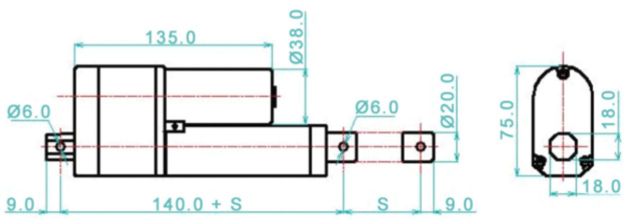
Keeping in mind problem specifications we chose a structure which blend together all key elements and that respond to our request in the best way. Main properties that we want from our robot is softness so we decided to use VSA actuators and Pisa/IIT SoftHand, build a robot with only these motors is not easy because they have torque boundaries which pose constrain on chosen configuration, moreover *end-effector* weigh about 0.5 kg so some motors endure high torques.

Although we want a robot with soft characteristics it is not necessary that all motors are VSA, but only actuators which interact directly with extern should, so we resolve one of main problem using classic motors at begin of chain; torque produce by these motors is not enough for pull up entire structure so we added gravity compensator mechanism to contrast gravitational pull.

Gravity compensator are composed by two sprint at constant coefficients, we posed them two for side to help motors eliminating structural weigh.

In the whole robot we count seven actuators: at begin of chain we find three motors which are posed orthogonally one to another and they shape a three-axis Cartesian structure, mount on last prismatic we mounted the following three VSA motors which permit to change attitude of the gripper, last actuator is the Pisa/IIT SoftHand which is structure *end-effector*.

(a)




(b)

Figure 4.1: (a) - Electric Piston , (b) - Electric Piston CAD.

We choose our configuration keeping in mind problem explain before so first motors is Z axis, which pull up all following part of the structure, second and third are respectively X and Y axis and they represent Cartesian robot.

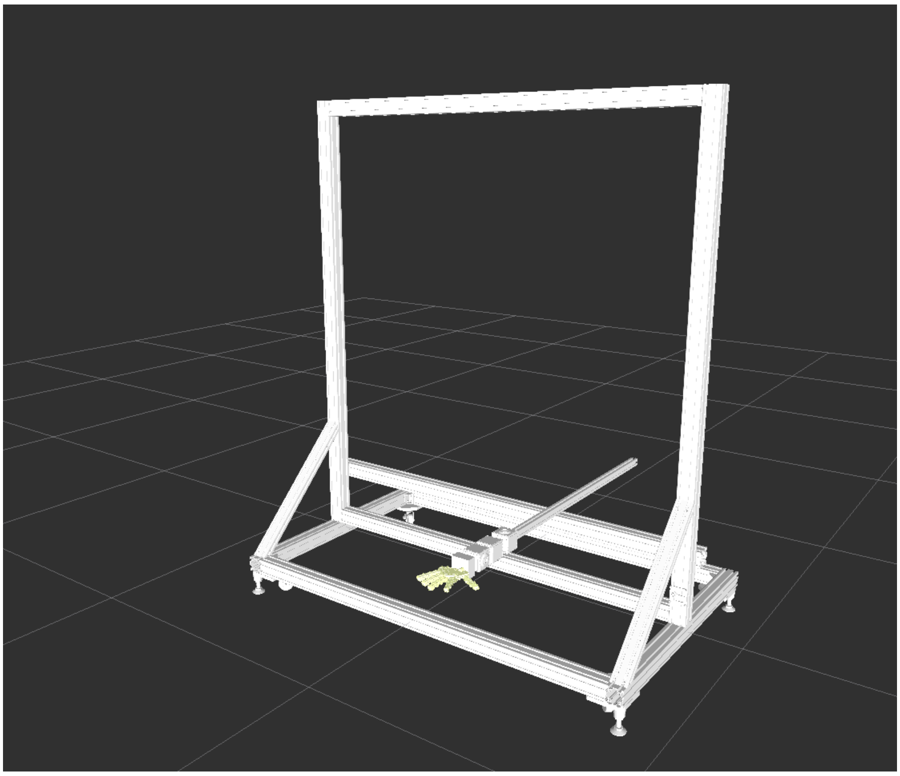


Figure 4.2: Structure Model on Rviz.

We pose a specific attention on how to choose configuration of three VSA to get maximum movement without overstep torque boundary, we decide to pose motors in serial configuration in Yaw, Roll and Pitch respectively fourth, fifth and sixth motors.

We reduce Pisa/IIT SoftHand operating space but we should not concern about torque anymore, this trade off between configuration and movements is not very constraining and it allows to reach all positions and attitude , in Fig.4.2 is shown the described structure.

Due to torque and dimensions constrains we decided to add a further tools to our structure which help us to hold object during grasp and at the same time extend grasp range of gripper, moreover, under specific situation, it permit us to reach grasp more robust than with only Pisa/IIT SoftHand, last event is common in particular situation where object orientation is not easy to grasp or it is not possible change actual orientation due to presence of environmental constrain.

In Fig.4.1(a) is shown the electric piston used, it can extend up to 33 cm its length and we mount at the end of tools an *ad-hoc* 3D printed flat, furthermore we upholstered it with neoprene to improve outside friction.

## 4.2 Pisa/IIT SoftHand

Pisa/IIT SoftHand represent an original solution of Soft Robotics, it is an anthropomorphic gripper extremely under actuated which control nineteen DOF of an human hand with only one motor.

Concept behind Pisa/IIT SoftHand is the synergies, researchers studied human grasp on different objects dimension and shape in details and after these study we discovered several grasp paths, called synergies, moreover we discovered that three of them recur frequently and they together cover most of the entire possible grasp situations [All14].

Figure 4.3: Pisa/IIT SoftHand.

Pisa/IIT SoftHand based is motion on tendons, a cable run for all hand and it is tied to a motor, when actuator begin to turn the tendon roll on motor pulley and the hand close, when cable is released the gripper back open.

Depending on how tendons pass through the Pisa/IIT SoftHand we can reach different behaviors, we distinguish two important kind of gripper: Pinch Grasp and Power grasp; they represent two different kind of closure, first endorse index finger and thumb closure first whereas Power grasp type has a surrounding approach of hand center. It is necessary highlight that we can't pass directly from one to another configuration because a configuration change impose a new rooting of tendons inside the hand.

Depending on situations a type is best than other and *vice versa*, but both of them have adaptive properties, in fact when gripper meet an obstacle the free parts keep moving changing the global grasp, thanks to that Pisa/IIT SoftHand adapt itself on object and permit strong, speed and easy grasp.

Another important aspects about Pisa/IIT SoftHand is robustness, each finger is composed by a series of disjointed parts linked by tendon together so we can warp all elements without any problems,

During our experiment we take in account Pisa/IIT SoftHand in *pinch grasp* configuration.

## 4.3 qbmove

Nowadays only few companies manufacture of electric motors has VSA in his own sales catalogs, there are several reasons about that but one of most important is inertia of industrial sector to change in fact factories endorse robustness instead technology innovations.

Nevertheless, new technologies take a lot of interesting improvements which are useful to common industry, furthermore recent study take on a new challenge, human safety during motor operation.

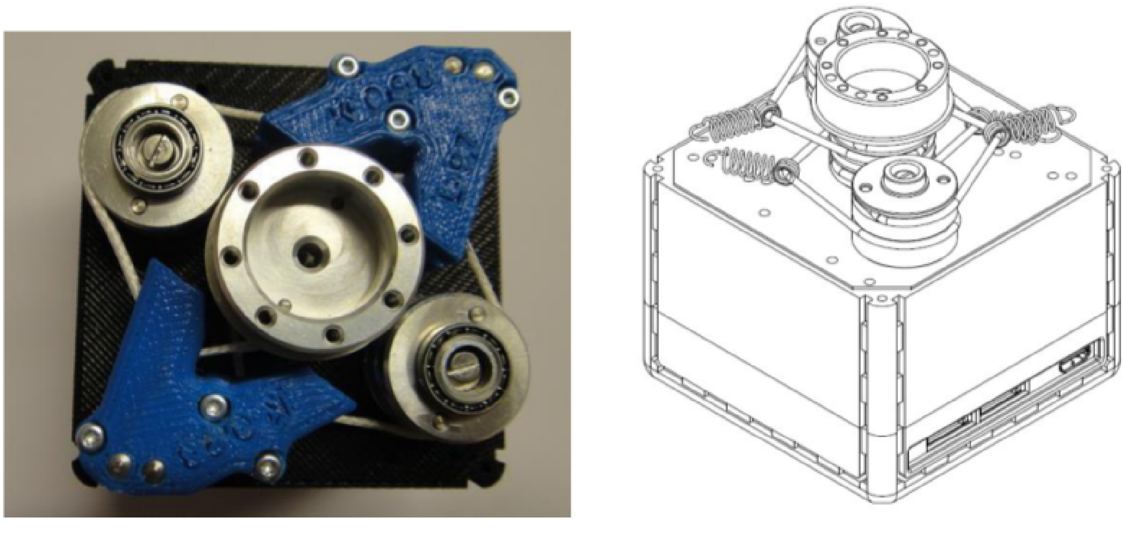
Between all this news one of them are VSA motors, we decided to use a specific VSA motor [All11] designed by Research Center E.Piaggio and produced by a spin-off of this center, qbRobotics. This company decide to keep almost everything, hardware and software, open-source so we could investigate or modify all aspects about motors chosen.

In first approximation we can see this kind of VSA motor like a mass, spring e damping serial which represent a mathematic simplification, basically the device use two independent motors link together by way of pre-tensioned cables constrained to a spring with known load coefficients.

Thanks to this structure, when an exogenous torque is applied, the response of shaft depends on the two motors configuration, moreover we can set different configurations continuously through a specific input.

Figure 4.4: Motore qbCube.

### 4.3.1 Mechanics



(a) (b)

Figure 4.5: (a) - Real image of motor, (b) - CAD of the motor.

How we touched upon, inside the cubical chassis are mounted two motors, MODELLO produced by, which have a tension operating range from 8 V to 25 V giving, individually, a torque of about 1.2 N m.

On the head of each motors is fixed a rotating roll which turn in-build with it, the roll pair are connected through a cable, carefully pre-tensioned, and restricted by springs.

Cables transmit the movement of two motors to center shaft which is the only part directly connect with other outside devices, this specific kind of configuration gives to the motor a particular rotating capacity which depend on symmetric movements of each single motors.

The different way of motors command represent an extra degree of freedom respect traditional devices, and we are able to drive the stiffness of outside motor, in this way we can have different response to a external torque depending how we set this further input.

We can drive internal motors in two different ways:

- Reflecting movement, $\delta_1(t) = \delta_2(t)$
- Out of sync to precise angle, $\delta_1(t) = \delta_2(t) + \Delta(t)$

It is easy see that the second method represent a general form of problem and the first is only a specific situation where offset is not zero. $\Delta$ inside formula is called *preset* and it is physically related to stiffness of motor shaft.

Explaining how *preset* influence stiffness on the shaft we make an example, suppose to set $\Delta = 0$, so both actuators chase the same position and offset between them is zero (when no torque are applied); when an exogenous input apply a torque on shaft it rotates, motor support movements which is damped by spring and thanks to them internal motors positions don't change.

But $\Delta$ is a dynamic input which can be modulated continuously, so when we increase this value motors make a tension between internal actuators already in external zero torque condition, thanks to this *preset* shaft get less range when a exogenous torque is applied than before. Keep going on this way we stretch spring between motors until maximum extension, this value represent max *preset* available of outside shaft and it depends on spring coefficient, when we set $\Delta = \Delta_{max}$ stiffness reach its own most.

A qbmove, which set *preset* on maximum value, has the same operating principle of a classic motor, this assumption is only a model because a lot of factors affect the operation: motors play, springs asymmetrical, asymmetric pre-tensional cables, and so on.

### 4.3.2 Motor Characteristic

Data-sheet of motor give us further informations about how device works, in Fig. we can observe the preset/torque relationship which is strongly incisive but, at same time, it has a non linear function.

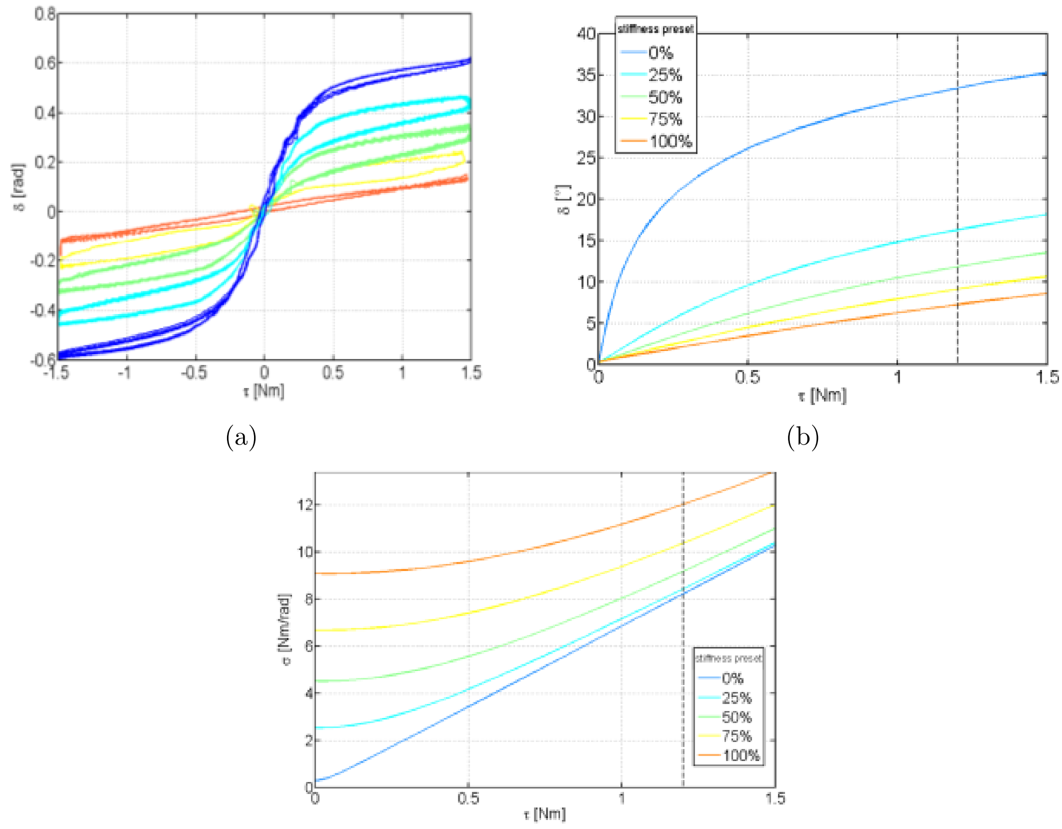
Motor behavior represent a crucial point when we design controllers with this kind of devices, so we should include behavior inside the process of control design.

Fig. 4.6(b) show what we explain previously, when we move external shaft providing a torque, at max 1.2 N m, deflection depends on set *preset* in a strongly way. When we maximize the *preset*, red line, motor has an almost linear behavior and when it decrease the not linearity increase in a exponential way.

### 4.3.3 Electronic Characteristic

Qbrobotics provides qbCube with a PCB owner, but at the same time they release their *firmware* under *open-source* license and thanks to that anyone can customize his own software.

On the electronic card we find a PSoC ®3.0 which mount 8051 microprocessor manufactured by Keil, moreover it is supply with several sensor pins usually used to connect board with motors *encoder* or other applications.




(a) (b) (c)

Figure 4.6: (a) - Characteristic Deflection-Torque, (b) - Characteristic Deflection-Torque depending on *preset*, (c) - Characteristic Stiffness-Torque

When board is used inside qbcube three sensor ports host the *encoders*, in particulars one for external motor shaft and two for inside motors, otherwise these pins can be set properly with a custom *firmware*.

qbCube has a lot of features which permit a set of different application, the board scale the power tension to 8 V irrespective of supply source power, this operation is possible thanks to dynamic PWM changing which command motors, moreover we can measure the absorb current from each motors.

Manufacturer fix the firmware speed at frequency of 1 kHz so we can manage a new fully operations (get measurement, get currents and set position) turn every 1 ms.

We can link more VSAs together by an ERNI cable and the paradigm followed is RS485 protocol approach which is a differential communication model based on ID, in the end we can interact with the chain by a micro USB cable.

An important problem on devices is speed communication, in fact it represent a strong limit in some situation for performance, manufacturer establish a serial communication with BaudRate of 460800 which represent the speed transmission for a single symbol, it is easy understand that time needed for a communication cycle depends on number of devices used, when this limit is exceed we start lost data proportionally.

Data-sheet suggest to stay in order of 500 Hz for each qbCube but this value is approximative and depends on chosen configuration so could be strong different in some case.

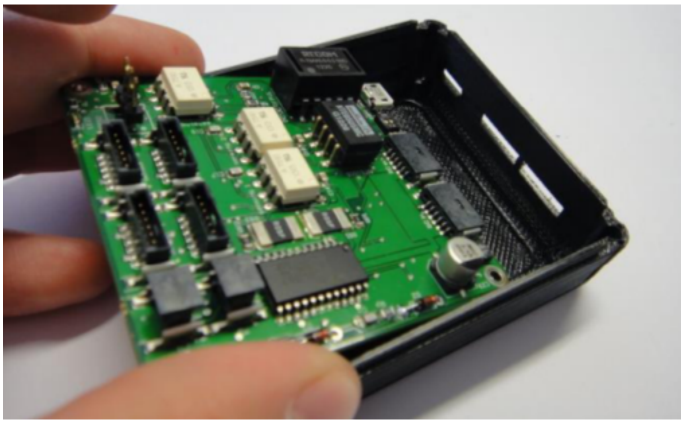


Figure 4.7: Scheda elettrica *qbcontrol_beta4.2*.

### 4.3.4 Embedded Control

Every single device have a feedback loop control which represent the strongest relationship between hardware and software in all structure.

The two motors inside qbCube are controlled with a PD controller which permit to reach the reference position request from outside to each single motors.

Default set coefficients are: nevertheless we can change these values when a special

| P | I | D |
|---|---|---|
| 0.1 | 0 | 0.8 |

Table 4.1: Coefficienti PID motori.

behavior is request, parameters are simply tunable through an *ad-hoc tool* for *ad-hoc* setting them.

Controller actions on the two motors have the same coefficients but they keep separate internal work, in spite of the single behavior motors act together to move central shaft.

## 4.4 Perception

We need a vision system to get informations about work scene which surround us and also we want determinate a specific object inside the scene to grasp. A lot of recognition algorithm exist in literature which use only 2D image but we need also informations about object position, orientation and dimensions, in fact these data help us on grasping process, because of these request we chose a 3D vision camera common in research field.

Asus XtionPRO, Fig.4.8, give us a, so call, point cloud composed by a series of structures, each of them include a three-dimensional points respect a given reference tern, usually inside camera, and a RGB information about color read from a RGB sensor in that specific point.

In this way we have information about what surround us and how far object are, point cloud are elaborate through algorithms which analyze scene and get out informations.

We decide to mount camera directly on an actuator so its reference tern is moved in-build with Cartesian structure, in this way we get data every time we need without potential vision occlusion.


Figure 4.8: Asus XtionPro 3D camera.

## 4.5 Static and Dynamic Analysis

We analyze structure movements with mathematical tools, first step we take advantage of mechanical properties and separate analysis in two part:

1. At the begin we have three prismatic joints which permit Z, X, Y movements respectively.
2. Last three are rotate joints which permit Yaw, Roll e Pitch rotation respectively.

Structure DOF are six which cover all possible movements and orientation in free three-dimensional space, in fact Jacobian matrix is a $6x6$ with full rank, it is clear that when we use robot in a environment the actual constrains reduce DOF.

We followed a standard parametrization of robot based on joints and links call *Denavit-Hartenberg*, in Tab.4.2; from this table we calculate Jacobian matrix which permit us to understand how to move structure to reach a specific position and it connect configuration space with three-dimensional space.

| Link | $a_i$ | $\alpha_i$ | $d_i$ | $\theta_i$ |
|---|---|---|---|---|
| 1 | 0 | $\frac{\pi}{2}$ | ${d_1}^*$ | $\frac{\pi}{2}$ |
| 2 | 0 | $\frac{\pi}{2}$ | ${d_1}^*$ | $\frac{\pi}{2}$ |
| 3 | 0 | $\frac{\pi}{2}$ | ${d_1}^*$ | $\frac{\pi}{2}$ |
| 4 | 0 | $-\frac{\pi}{2}$ | 0 | ${\theta_4}^* + \frac{\pi}{2}$ |
| 5 | 0 | $\frac{\pi}{2}$ | $c_4 + c_5$ | ${\theta_5}^*$ |
| 6 | 0 | $\frac{\pi}{2}$ | 0 | ${\theta_6}^*$ |

Table 4.2: Denavit-Hartenberg Table.

$$J = \begin{pmatrix} 0 & 0 & 1 & J_{14} & J_{15} & J_{16} \\ 0 & -1 & 0 & J_{24} & J_{25} & J_{26} \\ 1 & 0 & 0 & 0 & -\mathrm{a5}\sin(\theta 5)\sin(\theta 6) & \mathrm{a5}\cos(\theta 5)\cos(\theta 6) \\ 0 & 0 & 0 & 0 & \cos(\theta 4) & -\cos(\theta 5)\sin(\theta 4) \\ 0 & 0 & 0 & 0 & \sin(\theta 4) & \cos(\theta 4)\cos(\theta 5) \\ 0 & 0 & 0 & 1 & 0 & \sin(\theta 5) \end{pmatrix}$$

$$J_{14} = (\mathrm{a3}+\mathrm{a4}+\mathrm{a5}\cos(\theta 6))\sin(\theta 4)+\mathrm{a5}\cos(\theta 4)\sin(\theta 5)\sin(\theta 6) \qquad J_{15} = \mathrm{a5}\cos(\theta 5)\sin(\theta 4)\sin(\theta 6)$$

$$J_{16} = \mathrm{a5}(\cos(\theta 6)\sin(\theta 4)\sin(\theta 5) + \cos(\theta 4)\sin(\theta 6))$$

$$J_{24} = \mathrm{a5}\sin(\theta 4)\sin(\theta 5)\sin(\theta 6) - \cos(\theta 4)(\mathrm{a3} + \mathrm{a4} + \mathrm{a5}\cos(\theta 6))$$

$$J_{25} = -\mathrm{a5}\cos(\theta 4)\cos(\theta 5)\sin(\theta 6) \qquad J_{26} = \mathrm{a5}\sin(\theta 4)\sin(\theta 6) - \mathrm{a5}\cos(\theta 4)\cos(\theta 6)\sin(\theta 5)$$

$$J^{-1} = \begin{pmatrix} 0 & 0 & 1 & J^{-1}_{14} & J^{-1}_{15} & 0 \\ 0 & J^{-1}_{22} & 0 & J^{-1}_{24} & J^{-1}_{25} & J^{-1}_{26} \\ 1 & 0 & 0 & J^{-1}_{34} & J^{-1}_{35} & J^{-1}_{36} \\ 0 & 0 & 0 & \frac{\sin(\theta 4)\sin(\theta 5)}{\cos(\theta 5)\cos^2(\theta 4)+\cos(\theta 5)\sin^2(\theta 4)} & -\frac{\cos(\theta 4)\sin(\theta 5)}{\cos(\theta 5)\cos^2(\theta 4)+\cos(\theta 5)\sin^2(\theta 4)} & 1 \\ 0 & 0 & 0 & \frac{\cos(\theta 4)\cos(\theta 5)}{\cos(\theta 5)\cos^2(\theta 4)+\cos(\theta 5)\sin^2(\theta 4)} & \frac{\cos(\theta 5)\sin(\theta 4)}{\cos(\theta 5)\cos^2(\theta 4)+\cos(\theta 5)\sin^2(\theta 4)} & 0 \\ 0 & 0 & 0 & -\frac{\sin(\theta 4)}{\cos(\theta 5)\cos^2(\theta 4)+\cos(\theta 5)\sin^2(\theta 4)} & \frac{\cos(\theta 4)}{\cos(\theta 5)\cos^2(\theta 4)+\cos(\theta 5)\sin^2(\theta 4)} & 0 \end{pmatrix}$$

$$J^{-1}_{14} = \frac{\mathrm{a5}\cos(\theta 5)\cos(\theta 6)\sin(\theta 4) + \mathrm{a5}\cos(\theta 4)\cos(\theta 5)\sin(\theta 5)\sin(\theta 6)}{\cos(\theta 5)\cos^2(\theta 4) + \cos(\theta 5)\sin^2(\theta 4)}$$

$$J^{-1}_{15} = \frac{\mathrm{a5}\cos(\theta 5)\sin(\theta 4)\sin(\theta 5)\sin(\theta 6) - \mathrm{a5}\cos(\theta 4)\cos(\theta 5)\cos(\theta 6)}{\cos(\theta 5)\cos^2(\theta 4) + \cos(\theta 5)\sin^2(\theta 4)}$$

$$J^{-1}_{22} = \frac{-\cos(\theta 5)\cos^2(\theta 4) - \cos(\theta 5)\sin^2(\theta 4)}{\cos(\theta 5)\cos^2(\theta 4) + \cos(\theta 5)\sin^2(\theta 4)}$$

$$J^{-1}_{24} = \frac{-\mathrm{a5}\cos^2(\theta 4)\sin(\theta 6)\cos^2(\theta 5) - \mathrm{a3}\cos(\theta 4)\sin(\theta 4)\sin(\theta 5))+}{\cos(\theta 5)\cos^2(\theta 4) + \cos(\theta 5)\sin^2(\theta 4)}$$

$$\frac{-\mathrm{a4}\cos(\theta 4)\sin(\theta 4)\sin(\theta 5) - \mathrm{a5}\sin^2(\theta 4)\sin(\theta 6) + \mathrm{a5}\sin^2(\theta 4)\sin^2(\theta 5)\sin(\theta 6)}{\cos(\theta 5)\cos^2(\theta 4) + \cos(\theta 5)\sin^2(\theta 4)}$$

$$J^{-1}_{25} = \frac{\mathrm{a3}\sin(\theta 5)\cos^2(\theta 4) + \mathrm{a4}\sin(\theta 5)\cos^2(\theta 4) - \mathrm{a5}\sin(\theta 4)\sin^2(\theta 5)\sin(\theta 6)\cos(\theta 4)+}{\cos(\theta 5)\cos^2(\theta 4) + \cos(\theta 5)\sin^2(\theta 4)}$$

$$\frac{-\mathrm{a5}\cos^2(\theta 5)\sin(\theta 4)\sin(\theta 6)\cos(\theta 4) + \mathrm{a5}\sin(\theta 4)\sin(\theta 6)\cos(\theta 4)}{\cos(\theta 5)\cos^2(\theta 4) + \cos(\theta 5)\sin^2(\theta 4)}$$

$$J^{-1}_{26} = \mathrm{a5}\sin(\theta 4)\sin(\theta 5)\sin(\theta 6) - \cos(\theta 4)(\mathrm{a3} + \mathrm{a4} + \mathrm{a5}\cos(\theta 6))$$

$$J^{-1}_{34} = \frac{-\mathrm{a5}\cos(\theta 4)\sin(\theta 4)\sin(\theta 6)\cos^2(\theta 5) - \mathrm{a3}\sin^2(\theta 4)\sin(\theta 5) - \mathrm{a4}\sin^2(\theta 4)\sin(\theta 5)+}{\cos(\theta 5)\cos^2(\theta 4) + \cos(\theta 5)\sin^2(\theta 4)}$$

$$\frac{-\mathrm{a5}\cos(\theta 4)\sin(\theta 4)\sin^2(\theta 5)\sin(\theta 6) + \mathrm{a5}\cos(\theta 4)\sin(\theta 4)\sin(\theta 6)}{\cos(\theta 5)\cos^2(\theta 4) + \cos(\theta 5)\sin^2(\theta 4)}$$

$$J^{-1}_{35} = \frac{\mathrm{a5}\sin^2(\theta 5)\sin(\theta 6)\cos^2(\theta 4) - \mathrm{a5}\sin(\theta 6)\cos^2(\theta 4) + \mathrm{a3}\sin(\theta 4)\sin(\theta 5)\cos(\theta 4)+}{\cos(\theta 5)\cos^2(\theta 4) + \cos(\theta 5)\sin^2(\theta 4)}$$

$$\frac{+\mathrm{a4}\sin(\theta 4)\sin(\theta 5)\cos(\theta 4) + -\mathrm{a5}\cos^2(\theta 5)\sin^2(\theta 4)\sin(\theta 6)}{\cos(\theta 5)\cos^2(\theta 4) + \cos(\theta 5)\sin^2(\theta 4)}$$

$$J^{-1}_{36} = -(\mathrm{a3} + \mathrm{a4} + \mathrm{a5}\cos(\theta 6))\sin(\theta 4) - \mathrm{a5}\cos(\theta 4)\sin(\theta 5)\sin(\theta 6)$$

In addition to geometric position and orientation we want understand how structure can move, under this point of view other important elements go into game and we should evaluate inertial properties and Coriolis's effect, 4.1 represent the general dynamic equation, where

B is inertial matrix, C Coriolis's matrix, G gravity vector and $F_g$ are external forces and $\tau$ are external torques.

$$B(\dot{q})\ddot{q} + C(q,\dot{q})\dot{q} + G = \tau \tag{4.1}$$

$$B = \begin{pmatrix} B_{11} & 0 & 0 & 0 & B_{15} & B_{16} \\ 0 & B_{22} & 0 & B_{24} & 0 & B_{26} \\ 0 & 0 & B_{33} & B_{34} & 0 & B_{36} \\ 0 & B_{42} & B_{43} & B_{44} & B_{45} & B_{46} \\ B_{51} & 0 & 0 & B_{54} & B_{55} & B_{56} \\ B_{61} & B_{62} & B_{63} & B_{64} & B_{65} & B_{66} \end{pmatrix} \qquad C = \begin{pmatrix} 0 & 0 & 0 & 0 & C_{15} & C_{16} \\ 0 & 0 & 0 & C_{24} & C_{25} & C_{26} \\ 0 & 0 & 0 & C_{34} & C_{35} & C_{36} \\ 0 & C_{42} & C_{43} & C_{44} & C_{45} & C_{46} \\ C_{51} & C_{52} & C_{53} & C_{54} & C_{55} & C_{56} \\ C_{61} & C_{62} & C_{63} & C_{64} & C_{65} & 0 \end{pmatrix}$$

$$G = \begin{pmatrix} -9.81(\mathrm{m1} + \mathrm{m2} + \mathrm{m3} + \mathrm{m4} + \mathrm{m5} + \mathrm{m6}) \\ 0 \\ 0 \\ 0 \\ -9.81(-\mathrm{a5m5}\sin(\theta5)\sin(\theta6) - \mathrm{a5m6}\sin(\theta5)\sin(\theta6)) \\ -9.81\mathrm{a5m6}\cos(\theta5)\cos(\theta6) \end{pmatrix}$$

# Chapter 5

# Motion Planning and Control

Together with the discussed hardware we implemented a custom software which is responsible about motors control, motion planning, grasp strategies adopted and blend all information by devices together.

An overview of work is shown in diagram Fig.5.1 where, ideally, we summarized the entire software in a single whole logic block which takes informations from outside: 3D camera, request object and robot measurements; and elaborate them through several algorithms to grasp the specific requested object in the best way.

We can divide this first approximation scheme in further two macro-blocks, Fig.5.2 , one for perception area and one responsible for planning and control of robot; during this thesis we don't deal with the first block, because another part of our team focus on it and give us their results, thanks to *divide et impera* approach we can think about perception segment like a pre-given function which we integrated in our code.


Figure 5.1: General scheme.

Take on motion planning and control macro which represent our study area and we split it in further parts which help us to explain how our custom control works and give us an overall vision of adopted strategies, 5.2.

When perception algorithm identify a specific object gives position and orientation to next step, we join these informations together with the kind of object to look for and we decide the best grasp strategies between a set of known routines, each of them have different movements and generate several robot states to be followed; when routine finished our process a bunch of robot states are saved and the motion planning algorithm create a trajectory which interpolate them.

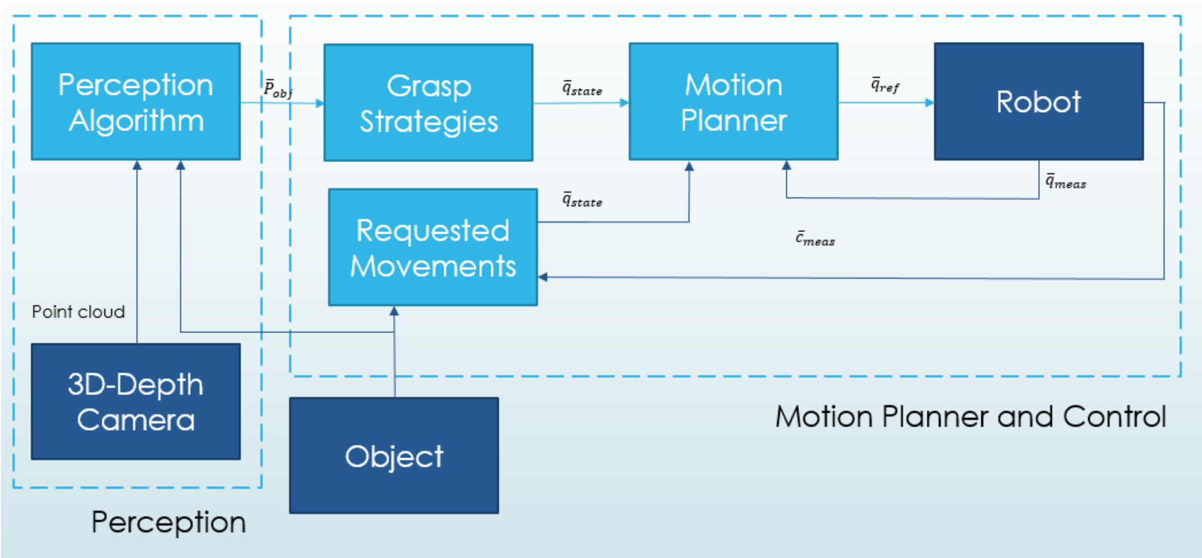



Figure 5.2: Macros scheme.

Motion planner output represent motor references which need a control to be reach perfectly, thanks to PID control all configurations are reached with low error rate.

Concerning grasp strategies we made a lot of experiments studying human grasp through Handle device, after that we analyze all data and categorized possible routines and implemented the code regarding each of them.

Thanks to this preparatory part we improved our knowledge about how bring object in environment constrained, we recall that our point of view was investigate the exploiting EC to pick up an object.

Motion planning method makes use of softness characteristic of motors so all referents points are not strongly establish but they depends on object properties and the environment where it is placed, a specific function permits this kind of operation by measuring actual deflection of the motors.

We automatized all operations so that robot performs task repeatedly after a wait period where receives a list of desired objects, thanks to ROS architecture different codes share information through ROS communication paradigm: topic and service; we decided to divide software in three communicating nodes, each of them write C++ language, in Fig.5.3 is shown nodes scheme and we pay a special attention about control node.

An internal thread is opened in control node which implements a PID controller on joints position, moreover it imposes Pisa/IIT SoftHand closure and open command.

## 5.1 Grasping strategies

As early discussed Pisa/IIT SoftHand is based on synergies and in our specific case we used a Pinch Grasp end-effector which best fit our requirements, although the device condenses a great grasp potentiality with an intrinsic simply application, Pisa/IIT SoftHand bring some problems about grasp approaching.

We investigated to this direction looking for an approach to solution of the problem, moreover we joined gripper characteristics with our robot structure which has some constrains about movements because it is composed by six actuators which give fully attitude and position control in a six dimensional space, but we made a step forward investigating

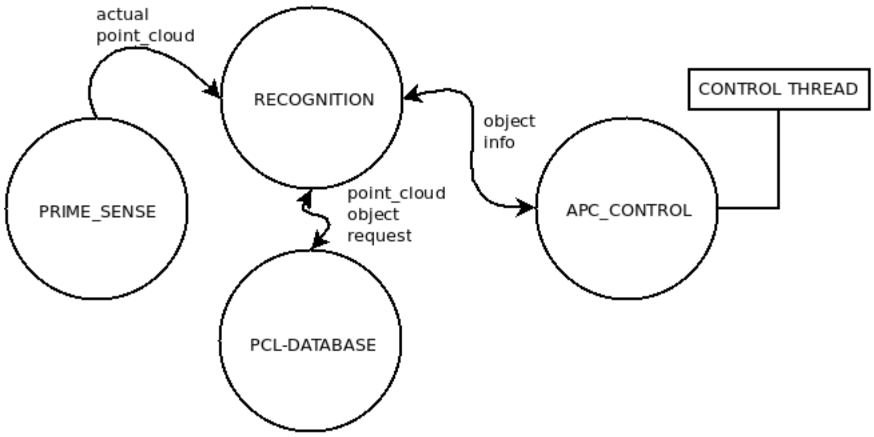



Figure 5.3: ROS working scheme.

over constrained environments.

Classic grasp methods analyze shape and dimensions of the object to extrapolate a set of minimum points which give the best grasp, but that process is very complex and, depending from situation, it can be onerous in time; in order of exploit soft robotics properties we design a grasp strategies composed by relative movements which exploit the actual EC in scene.

Usually camera measurements are uncertain and we know a blurred object position and attitude so when we approach the item it is not easy understand where every fingers are and often we need further sensors to know real position, another problem regarding uncertain, which can bring unpleasant results, happens when object is strongly flattened because motor try to reach reference position by any means, we avoided this issue through VSA.

Due to blurred measures the static movements don't look enough safe and we need a different approach at object, we can think a new kind of strategy first of all we decided the movement direction then robot changes position in that direction until an obstacle is reached, last part is directly connect with soft actuator which permit measurement of deflection on external shaft.

One of most interesting Pisa/IIT SoftHand properties is adaptability, thanks to its robustness under external pressure, which permit to deform structure without any damage for gripper or external elements, we use this ability in several situations:

- Take an object in the scene exploit EC
- Compensate uncertain on object position and attitude
- Isolate desired item from other

In the fist stage of our study we gather informations through experiments carry out by an human been with a specific device, call Handle, which permit to wear Pisa/IIT SoftHand.

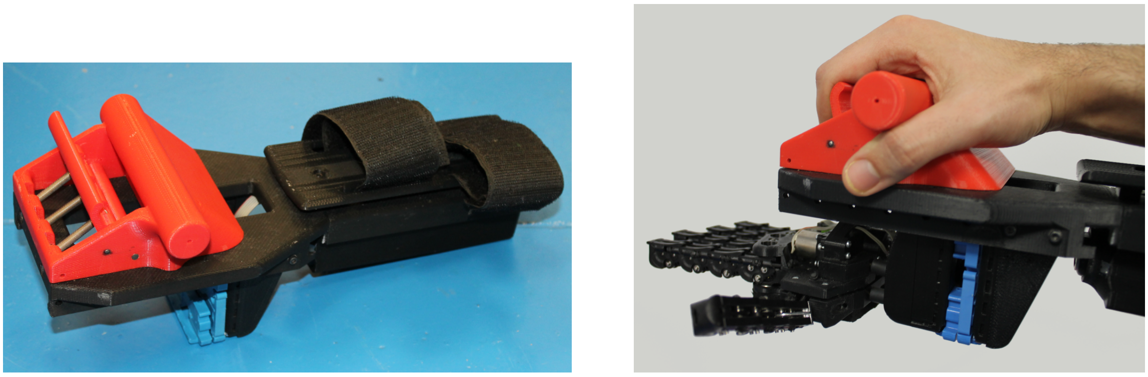
After that stage we analyzed data and clustered all find out grasp strategies in a groups, thereafter, following likelihood criteria, we sub-clustered beyond obtaining final set of grasp strategies.

### 5.1.1 Experiment with Handle

We made the experiments under known condition:

- Medium build tester, 1.80 m high.
- Environment constrained in five direction: bottom, top, left, right and end; a classic shelf 1.70 m high composed by bins.
- Split in three parts environment.
- Using an Handle device.
- Under a precondition set of objects

Experiments space is a very common situation where a person has to take an object inside a library or from the pantry, but we want use our results with a soft gripper so it would be more interesting solve the problem directly, thanks to a specific interface device called *Handle*, Fig.5.4(a); when we wear Handle we can substitute our hand with the Pisa/IIT SoftHand, it is fixed on user arm and, through a lever, we can modulate closure rate.



(a) (b)

Figure 5.4: (a) - Handle device, (b) - Handle with Pisa/IIT SoftHand.

We split environment in three sector, Fig.5.5, to investigate better every possible position and attitude depending on zone, in fact every bin has a specific size and Pisa/IIT SoftHand can not operate without constrain, so when an items is pose in a side sectors we can not use all gripper potentiality.

- Central zone
- Left side zone
- Right side zone

We took a specific set of objects which are several shape, dimensions, compactness, in this way we have a varied sample of items which cover a lot of possible properties, in Fig.3.5 are shown objects take in account during experiment. Each item is posed under experiment individually in every possible orientation and in every zone, all data are reported on a structure called *DataBase* where we saved experiments results and, furthermore, we used it to implement grasp strategies in our code.

As previously mention our experimental case regarded both object on the plan, one EC, than external sides of bin which are vertical constrain, the main reason behind these considerations is possible object position in a shelf; when we replace a book on a library we

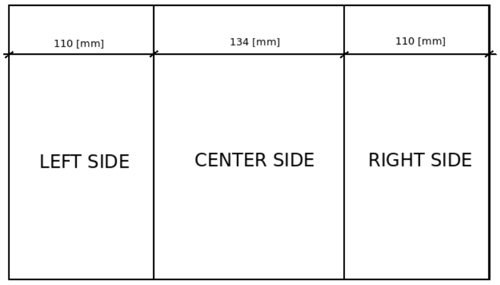



Figure 5.5: Environment zone splitting.

(a) (b)

Figure 5.6: (a) - Pick an object from Left Side , (b) - Pick an object from Center

pose it in vertical orientation, maybe with the book face aim external side, it is only one example of a set of natural gesture which are is very common in place inhabited by humans.

Nevertheless robot has to learn how interact with object, pose in specific position and orientation, in a boundary environment, we can find items near sides even frequently when a little space is filled with objects, this situation needs specific algorithm which take care of the event.

### 5.1.2 Grasp strategies Classification

Every object has a properly grasp strategies which depends on item shape, utilized gripper, position inside bin and other stuff in the near area; of course this assumption keeps truly also in our case but, after we observed all experiments, we discovered likelihoods between approach strategies implemented by user.

Some properties of approach was frequently, for example final gripper positions, orientations or wrist angle; under this point of view we proposed a classification which gather all experimented grasp in macro groups depending on these properties.

We proposed a total of seventeen classes, each of them has a specific implementations and it is used in more suitable situation, in Tab.5.1-5.2-5.3 are shown every strategies with its own explanation.

The set of strategies proposed is general and some of them are alike, so we decided to sharpen the groups and search connection between them, thanks to this second stage implementation was simpler because we can join two strategies together which composed a third, an example is CS and CP: the only difference between them it is a left movement so we can execute CP routine, move Pisa/IIT SoftHand to left and after all push down to grasp the object.

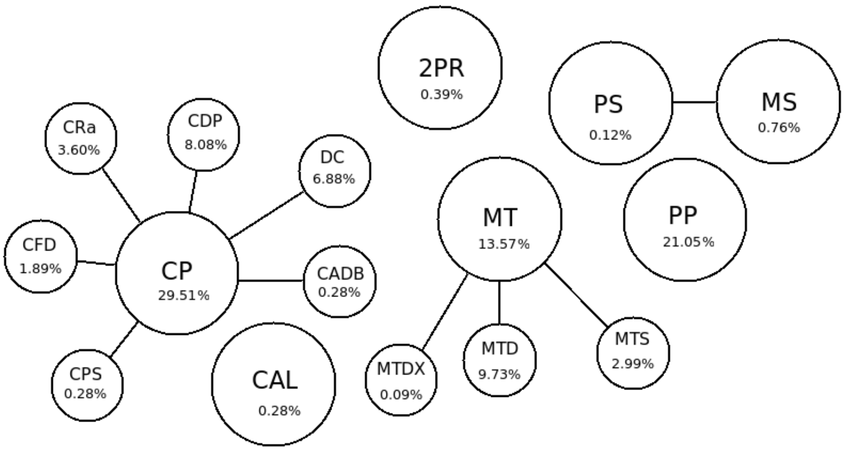



Figure 5.7: Group classification.

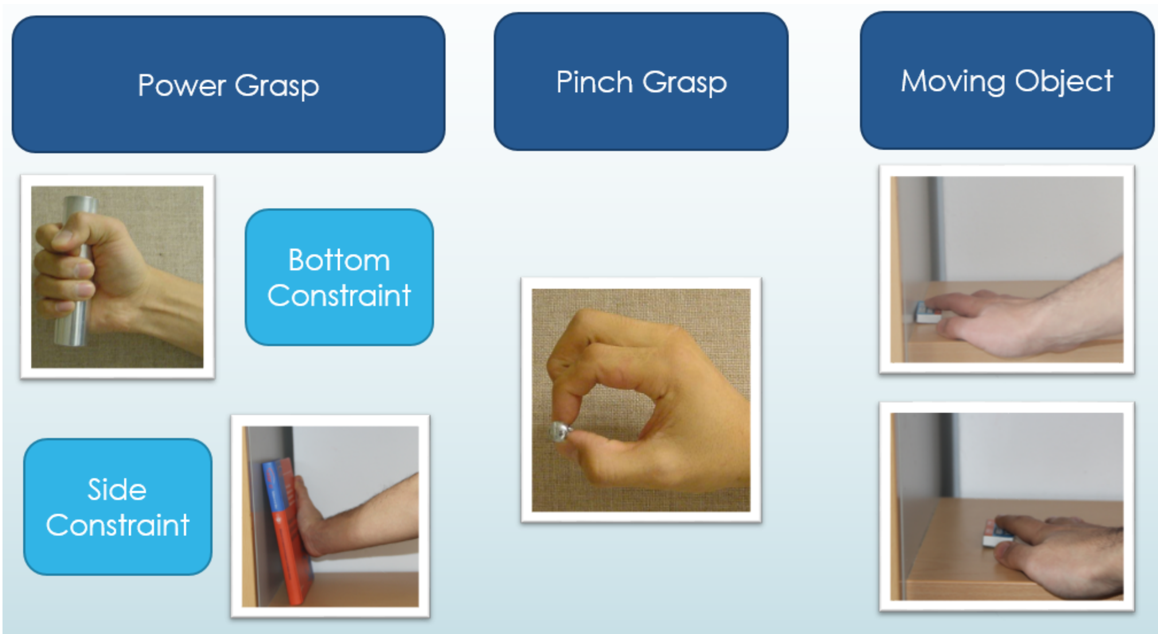



Figure 5.8: Sub-group classification.

| Strategies | Explanation |
|---|---|
| CP | Correspond gripper and object center, then push down |
| CR$\alpha$ | Correspond gripper and object center, turn *e-e* of $\alpha$ degrees, then push down |
| CFD | Correspond final part of gripper with object center, then push down |
| CS | Correspond gripper and object center, translate to left, then push down |
| CD | Correspond gripper and object center, translate to right,then push down |
| CADA | Correspond gripper center with and object right up corner,then push down |
| CADB | Correspond gripper center with and object right bottom corner,then push down |
| CASA | Correspond gripper center with and object left up corner,then push down |
| CASB | Correspond gripper center with and object left bottom corner,then push down |
| DC | Correspond the middle part of gripper with object center, then push down |
| CDP | Correspond the median point between thumb and index finger with object center, then push down |
| DCR | Correspond gripper and object center, push down, turn wrist of $\alpha$ |
| PP | Correspond final part of gripper with final part object, use piston, push down |

Table 5.1: Grasp strategies in Center zone.

Diagram Fig.5.8 show the news strategies and it disclose the discovered connections.

Thanks to gathered data we perform further analysis which revel a special relationship between object dimension and the grasp strategies adopted, in Fig.5.9 we can observe relationship percentages, from object set, classes and bin zone.

| Strategies | Explanation |
|---|---|
| CP | Correspond gripper and object center, then push down |
| CR$\alpha$ | Correspond gripper and object center, turn *e-e* of $\alpha$ degrees, then push down |
| CFD | Correspond final part of gripper with object center, then push down |
| CD | Correspond gripper and object center, translate to right,then push down |
| CADA | Correspond gripper center with and object right up corner,then push down |
| CADB | Correspond gripper center with and object right bottom corner,then push down |
| DC | Correspond the middle part of gripper with object center, then push down |
| 2PR | Double grasp |
| MT | Turn gripper at 90°, push left |
| PS | Correspond thumb with object center, push down, move object faraway from side |
| PP | Correspond final part of gripper with final part object, use piston, push down |

Table 5.2: Grasp strategies in Left zone.

## 5.2 Motion Planner

Every robot require a motion planner which is responsible for several operations as trajectory generation and collision avoidance; nowadays a lot of motion planner algorithms are implemented which have interesting features and, usually, an high integration and optimization of the code, *moveIt* is an example.

Although solutions of this problem are already presents we decided to write our own motion planner, because most of existing planning algorithms look at environmental constrains as obstacles and they don't exploit EC in the scene in any way, when structure touch an external element advise is launched to signal a collision.

But it is not all, we want implement specific grasp strategies which could be composed by two or more actions so it is hard, for an external motion algorithm, manage a sequence of actions in a little span of time.

Thanks to robot structure we can move last three joints and Pisa/IIT SoftHand directly in Cartesian coordinates and we manage attitude of *e-e* during approach session.

In Fig.5.10 is shown motion planner flowchart which is divided in grasp strategies, requested movements, trajectory generator and control loop.

We structured motion planner algorithm through configuration state vectors, each of them bring several informations inside as desired positions, moving method and even hand closure command, so a single state gives the references for each actuators and the way to reach it.

When the series of configuration is fully saved we interpolate all states and execute the develop trajectory, it is important stress a concept: every configuration is reached, apart from eventually small errors on measurement.

*Grasp strategies* and *Requested movements* blocks are the main contributors to generate states and, together, they represent the brain behind motion planning algorithm.

Last but not less important is *Control Loop* block, when motion planner set a motor reference we need a position control loop to bring actual motor angle to desired one, we implemented a PID controller which loop autonomously on a specific thread.

| Strategies | Explanation |
|---|---|
| CP | Correspond gripper and object center, then push down |
| CR$\alpha$ | Correspond gripper and object center, turn *e-e* of $\alpha$ degrees, then push down |
| CFD | Correspond final part of gripper with object center, then push down |
| CS | Correspond gripper and object center, translate to left, then push down |
| CASA | Correspond gripper center with and object left up corner,then push down |
| CASB | Correspond gripper center with and object left bottom corner,then push down |
| DC | Correspond the middle part of gripper with object center, then push down |
| CDP | Correspond the median point between thumb and index finger with object center, then push down |
| MTD | Turn gripper at $-90°$, push right |
| MS | Correspond the little finger with object center, push down, move object faraway from side |
| PaDP | Reach maximum right position available, push down |
| PP | Correspond final part of gripper with final part object, use piston, push down |

Table 5.3: Grasp strategies in Right zone.

### 5.2.1 Trajectory Generator

*Trajectory Generator* takes all saved state vector and interpolate them in a trajectory that join all points, as cited before state contain further information about moving constrain and tunable options.

A single state vector contains:

1. Robot Configuration
2. Tunable Presets of VSA motors
3. Pisa/IIT SoftHand Closure Rate
4. Kind of movement selected
5. Span of time to be waited
6. Rate value

A bunch of robot configurations composed the signal reference to be send to motors, but reference has a square wave shape that presents a lot of discontinuity points; robots need a smoother commands to avoid control problems and to eliminate peak of current when it is in motion.

One of most common solution to this issue is pass through a filter which transforms reference signal to a new one smoother than before, moreover this step get a fluid movements between state vectors.

Thanks to rate value option we can established the step time between two beside references, this value affects how robot follow reference and it represents a trade off between execution speed and eventually overshooting that control is not able to soften; because of that reasons we keep rate values about 0.125 which represent a good choice for our experiments, in Fig. 5.11 we can see the same signal with different rate values.

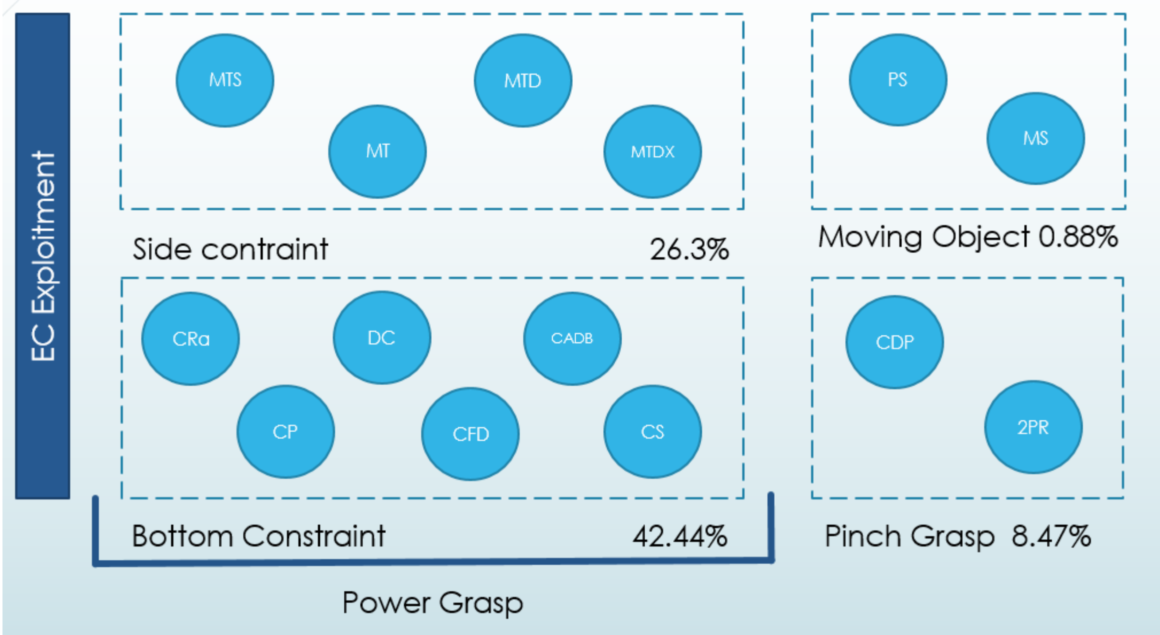



Figure 5.9: Grasp class statistics depend on pose.

We imposed that joints reach next state all at same time, so the motion speed are different even in the same vector state, in fact motor with long range go faster than joints which move almost anything, in this way there are not misalign during motion.

$$e_i(k) = r(k) - y_i(k-1), \quad \forall i \in \{1 \to 6\} \tag{5.1}$$

After we evaluate errors for all actuators we find the maximum value over them and we scale the left errors to slow their motion, moreover rate value is considered on calculation as a further factor element which modulate speed.

A lot of several interpolation methods exist in literature, but look at our robot characteristics we needed a method which permit us to cover long distance in small time when it is indispensable or use a linear interpolation when request, so we proposed two different ways, the difference between them is velocity profile:

- Constant speed profile
- Sigmoid speed profile

Both methods respect constrain of arrival time previously defined but the first one keep speed constant during all motion, in Fig.5.12(a) is shown an example of how it work and profile velocity diagram; the second one methodology impose a different speed during arm motion, following a sigmoid shape, at the beginning and at the landing robot moves slowly while in the middle part velocity increase, through this method we obtained a very fluid trajectory, we show in Fig.5.12 the relative scheme.

The sigmoid function implemented is:

$$y_i(k+1) = y_i(k) + \frac{e_i(k)}{1 + \exp^{-0.3t}}, \quad \forall i \in \{1 \to 6\} \tag{5.2}$$

$$y_{ij}(k+1) = y_i(k) + \frac{e_i(k) * j}{\max_{i=1 \to 6} e_i}, \quad \forall i \in \{1 \to 6\} \tag{5.3}$$

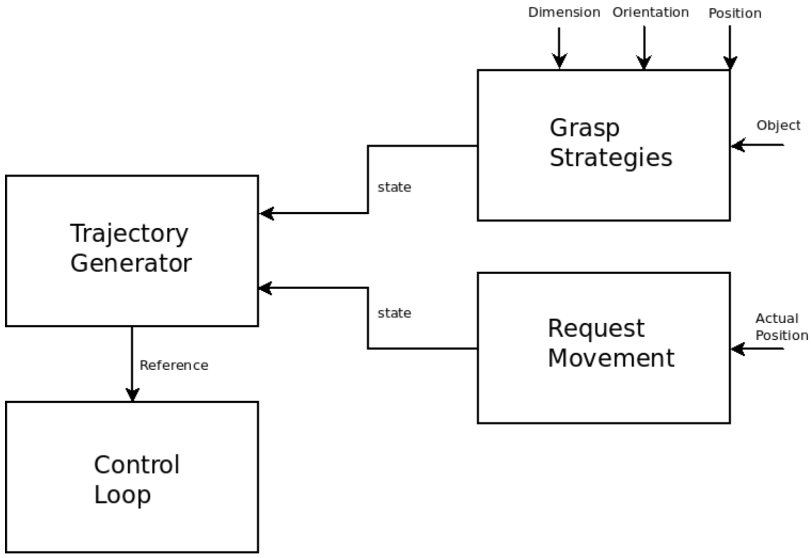



Figure 5.10: Motion Planning flowchart.

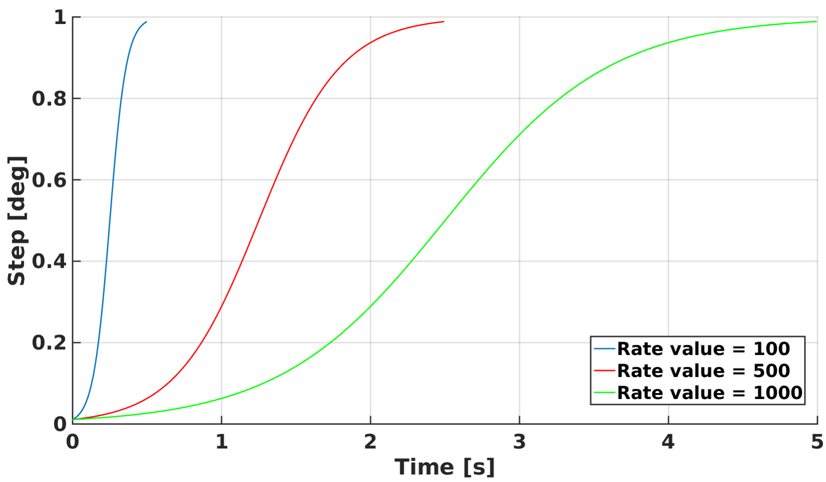



Figure 5.11: Step on rate value modulation.

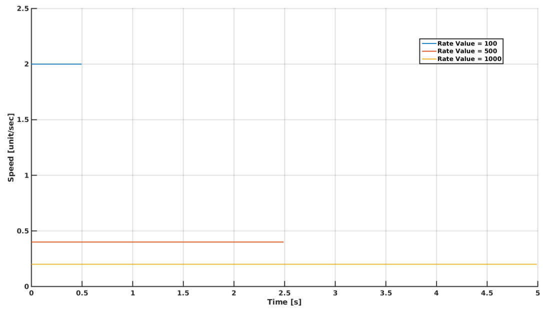
Furthermore we can set Pisa/IIT SoftHand closure rate thanks to a specific coefficients which represent percentage of closure, admissible values are from 0 to 1.

### 5.2.2 Motor Control

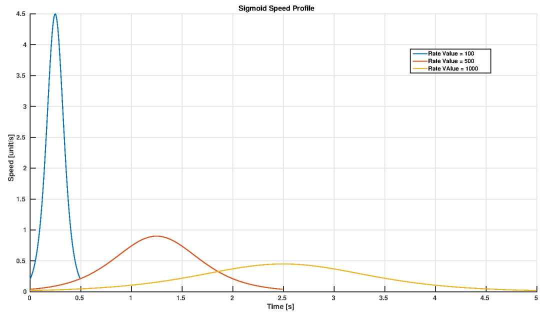
Bottom layer, Fig.5.13, represent the control on joints position, we recall that robot is composed by seven motors more further three which are installed on VSA to perform stiffness modulation.

1. Linear motors, the first three, have a PID control direct load on board firmware which regulate reference following.

2. Rotate motors, last three, are VSA and they have embedded controls which regulate positions of two internal motors, this configuration let shaft motor free to regulate stiffness.




(a)




(b)

Figure 5.12: (a) - Constant speed profile, (b) - Sigmoid speed profile.

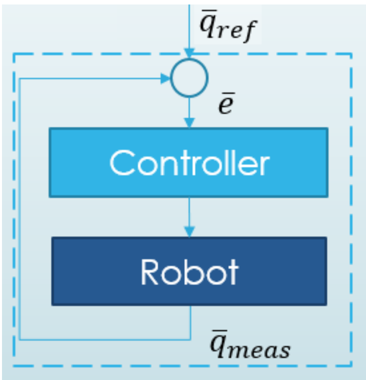



Figure 5.13: Control loop scheme.

3. Gripper Pisa/IIT SoftHand has only one motors which control hand closure, we can use a percentage to command actual closure.

| Actuator | $K_P$ | $K_I$ | $K_D$ |
|---|---|---|---|
| Linear 1 | 1 | 0.02 | 0 |
| Linear 2 | 0.7 | 0.0 | 0 |
| Linear 3 | 0.67 | 0.0 | 0 |
| VSA 4 | 0.07 | 0.7 | 0 |
| VSA 5 | 0.3 | 1.0 | 0 |
| VSA 6 | 0.05 | 0.6 | 0 |

Table 5.4: PID coefficients.

We tuned on board controllers of linear motors because we write a specific firmware for them, using a qbmove board, so they hadn't a factory embedded control already tuned; we let VSA controllers to default values.

Upon embedded layer we added a further control loop on shaft position of VSA actuators so, in this way, motors keep soft, on their limits, and at the same time it stay on requested position. This kind of control is necessary to compensate gravity, in fact it represent an exogenous torque that bring out from desired position the real one.

Ideally only one linear motor of three is affected by gravity because second and third are perpendicular to the ground, but robot structure is supplied by constant springs which compensate almost all the weight of other components, so linear actuators don't need any further control compensation.

Stiffness properties represent a very important point in our dissertation, so we support this feature through PID coefficients modification at *runtime*, values change during operation so that we can take benefit from different preset and at the same time we have not efforts motors and we keep small the power control request on each elements.

## 5.3 Perception

Together with main algorithm node our ROS scheme is composed by three further nodes, one for 3D camera that takes a three-dimensional picture of actual scene and saves image when a request is arrived, another node elaborate the point cloud to recognize a specific object inside the scene, finally last node stores all point clouds collected during a previous stage and used as comparison.

At the beginning of code the main algorithm communicate with perception layer through ROS service, we move robot on specific position and advise Asus xtionPRO node that we want take a picture, later the scene is saved on a topic and recognition node get scene from topic and start a cleaning stage.

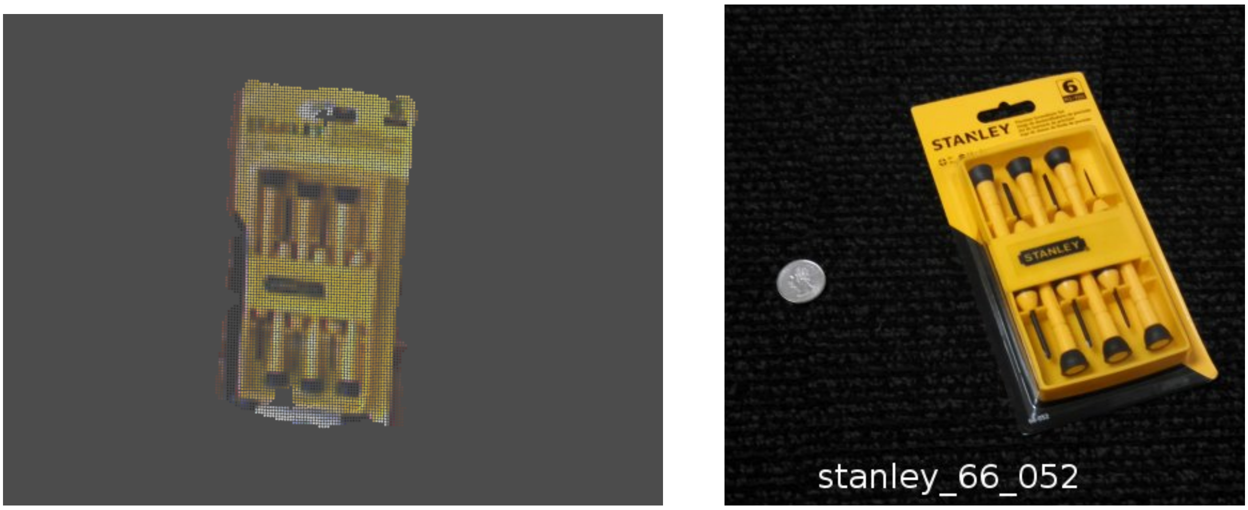
RGB-D camera takes a raw image that need to be clean through a filter algorithm that removes the outliers, the floor and other external environment planes, this part is called filtering After that image is clean we clustered the scene, so we obtained a finite number of small point clouds to be analyzed.

Recognition algorithm elaborates the remain clusters. It is composed of several sub algorithms that are implemented to improve recognition, for example a shape control is enabled together with a color threshold, this stage reduces the number of point clouds available to last step.

The survived clusters are preliminary scaled depending on the distance from the camera and after they are subjected to Iterative Closest Point(ICP) algorithm. It calculates the distance from actual cluster and the point clouds stored in the database, the smaller distance upon all the point cloud is taken as solution.

Information about object recognition are sent by ROS topic to the apc_control algorithm, but it is not the only data send to the motion planner.

We set the algorithm to send also the information about surrounding objects if any, because they are used in the decision process to manipulate the scene letting the desired object isolate. In particular, the left and right points are sent only when the distance from target and neighbor is with in a threshold.




(a) (b)

Figure 5.14: (a) - Stanley Point Cloud, (b) - Real Stanley image.

# Chapter 6

# Experimental Validation

## 6.1 Motor Control

Each motor was controlled by a PI controller and, after we set up them on the structure, finally we are able to investigate on response of these actuators directly on real structure.

At the beginning of test we posed all VSAs in their zero position, thereafter we set an input to the motors composed by a several kind of typical inputs: a ramp to 60°, a step to zero, a step to 15° and, after we bring again to zero, a step to 60° with a change of preset from zero to 30°.

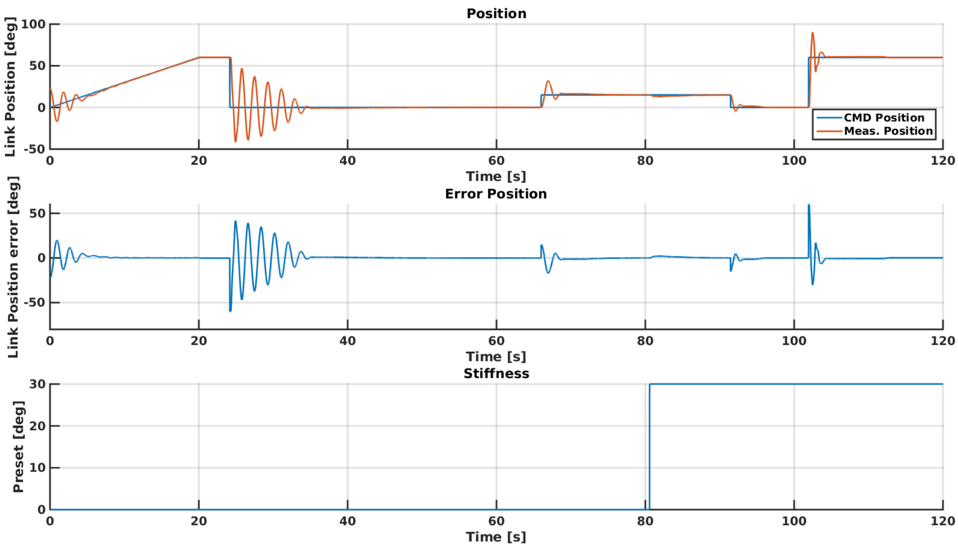



Figure 6.1: Yaw motor results.

Results shown a good response of motors even when subjected to big step, although these conditions are avoided from motion planner they gives us a further information about robot stability, we can think about the big changes like exogenous not requested inputs.

When preset is zero motors are completely soft and springs contribute to the little oscillation around reference position, fluctuation is proportional to external torque so Yaw actuator is more subjected because of the parts down-line of him.

Fig.6.1, 6.2, 6.3 show classical VSA work, in the first part of the test, preset is set to zero and oscillation are present but they are strongly reduce when preset reach an high value.

We can not talk of 30° like an absolute maximum preset, in fact each VSA has its own

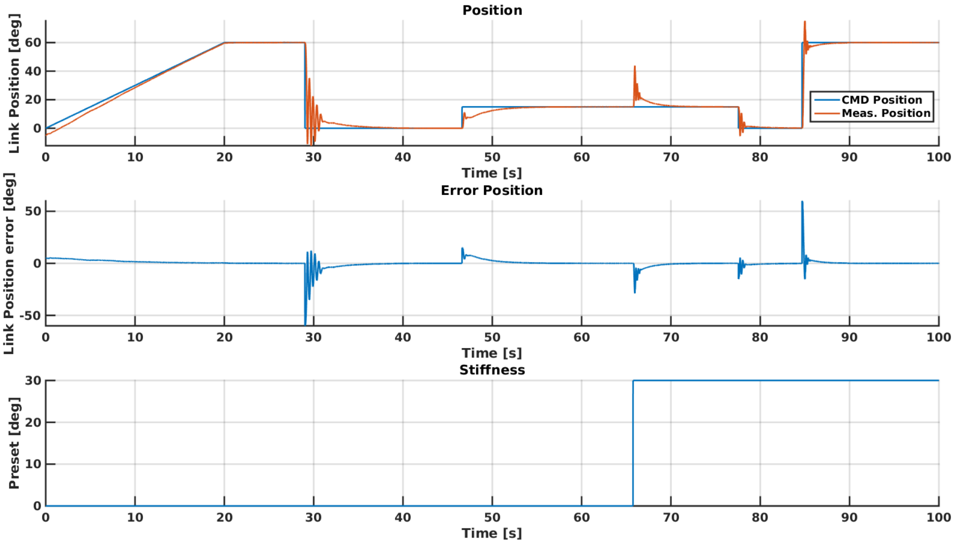



Figure 6.2: Roll motor results.

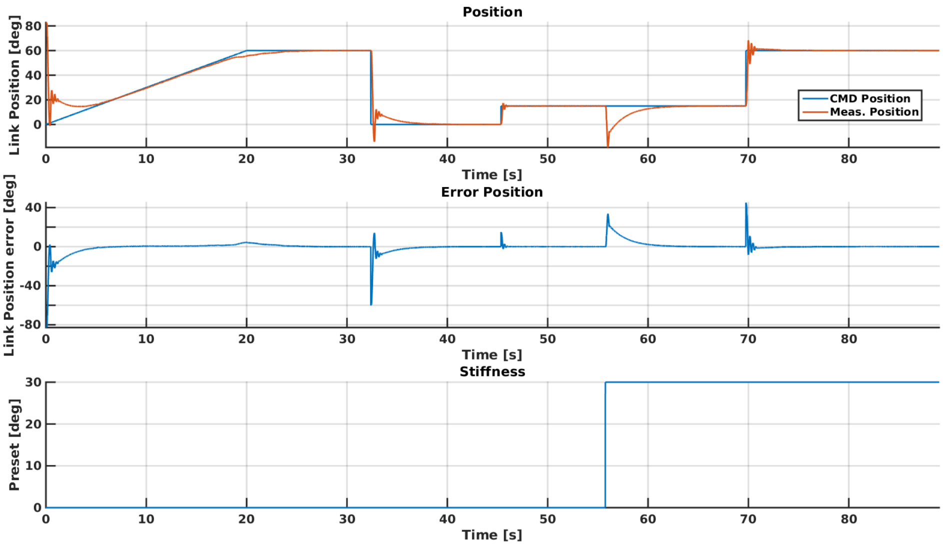


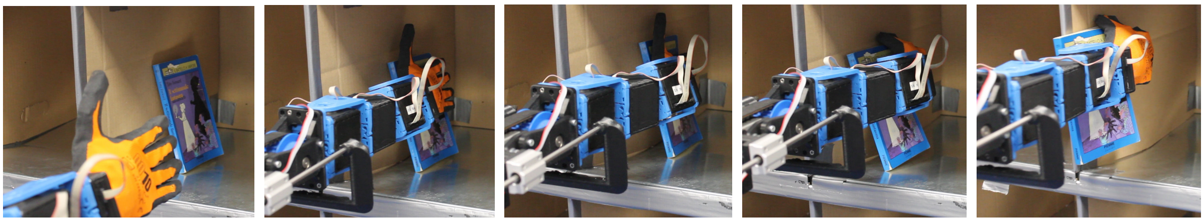

Figure 6.3: Pitch motor results.

maximum value depending on internal spring and linking cables, but a classic max preset of variable stiffness actuator is around 35°.

## 6.2 Motion Planner

We have already described how motion planner work and what kind of options are tunable, now we want test motion planner and gather results.

Motion planner algorithm is not responsible only of generate trajectory between two o more points, but it decide also which grasp strategy apply depending on data in the actual field, image sequences in Fig.6.4 show what happens after object informations are received from perception node and they are processed to get the best strategies.



(a) (b) (c) (d) (e)

Figure 6.4: Book Grasp photo sequence.

## 6.3 Case Study

Experiments illustrated that the exploited grasp strategies work fine when the objects are isolated. Items neighborhood can reduces the successful of the grasp routine because it obstacles the execution of the algorithm.

Under this point of view, an isolation algorithm is implemented to push away the near objects taking back the scene on a simpler situation where the require item is alone.

Isolation algorithm involves several issues, in particular two problems represent important study elements which cause complexity raising of the algorithms:

- Span of time requirement at runtime.
- Successful grasp checking.

Extreme cases support the previous explanation on isolation strategies, suggest issues are:

**Without *Span of time requirement at runtime* constraint**

The object point cloud is searched on a general point cloud of the bin took from RGB-D camera. After the recognition algorithm ends, the position and orientation of the object are send to motion planning algorithm, at the same time algorithm analyzes the neighbor object point clouds.

Items close to the target are pushed away from it and the scene becomes clean.

This methodology needs to recognize two or more objects, depends on the number of items in the bin, and furthermore needs to take out annoying objects and relocate them in a simpler position.

In this way the target becomes easy to grasp because potential occlusions are removed.

**Without constrain grasp successful check**

The simpler strategy is based on pushing of the objects that distances target from other items, this algorithm does not take in account shape of the near objects but only its location.

Recognition algorithm have not to recognize any other objects than target so the requested span of time is not increased, using this methodology can make situation worse because there are not way to know the new position of moved objects.

During a challenge the time is very important and, because of that, we preferred to use a third ways which is an hybrid strategy that mixes before two and consider the robot configuration available.

### 6.3.1 Target isolation strategies

The Pisa/IIT SoftHand is robust and adaptable to external pressure, when outside forces are imposed on the hand it can deforms its structure without damages for itself or the item.

Because of that we decided to hurt objects directly with the fingers and palms of the hand, the force imposed on the object push away the items.

Last problem to deal with is the possible situation where the moved object obstacles the target grasp, in fact the blind action can not guarantee the grasping successful.

Hybrid algorithm is located in the middle of the previously two methods, we decided to not discard all information provided by near *point cloud*.

Each cluster is analyzed to extract information where impose force to moves the objects away, we do not need to know what items is shown in point cluster but where should be moved.

The strategy can summarized as:

1. Recognize object in the scene
2. Identify further *point cloud* near the target
3. Analyze dimensions, shapes and structure of unnecessary clusters
4. Positioning and moving of near object
5. Grasping of the item

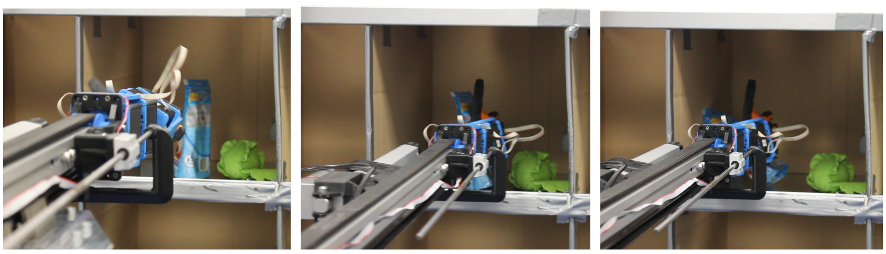
To increase the speed of the routine we decided that, depending on the grasp strategies, motion planner is advised of neighbors objects only when the grasp routine is affected by it.

*apc_ctrl* node takes informations from *recognize* node after that it completes all the recognition process; to reduce the number of information transmitted we select only positions of the adjacent objects and we send it.

When no items are present a zero vector is sent, this configuration is unique because it represents the origin of the camera frame and no objects can be locate there.

To push away an object inside the bin several observations are duty, the Pisa/IIT Soft-Hand has large about 20 cm, so, together with environmental constraints of the shelf, the workspace is narrow and an horizontal motion is not possible.

Instead if the wrist rotates the roll angle up to 90° or −90° the motion is possible inside the bin boundaries.



(a) (b) (c)

Figure 6.5: Push away algorithm photo sequence.

After that end-effector reaches the desired force point application, the fingers impresses a pressure divisible in two directions: to the final part of the bin and, in the same time, a translation of the object toward the extern, left or right.

In this way we distance requested object from not exploitable obstacles in the scene.

Finally when two objects are locate surrounding a target, the algorithm explained before is applied on both the objects and afterwards next step of the code is executed.

1

### 6.3.2 Calibration

Experiments and challenges provided a size for the environment where applied the motion planner and the grasp strategies.

Because of that we decided to write a specific code to calibrate movements depending on situations, in situations as the challenge this routine help user to set properly the robot in few time.

In the challenge we used Amazon like shelf but during experiments we have not an original standard Amazon shelf used in the delivery system so we use the measurements given by the Amazon site directly. A initial set up permit us to calibrate the entire structure in all possible situations (with Amazon-like shelf), in this way we are ready to start performing action on the new environment.

One possible way was use RGB-D camera point cloud to identify position of each bin depending on robot actual position, in fact understanding where limit sides of the shelf are, but that method can not calibrate saturation directly. Instead we implements an algorithm that calibrates saturation by deflection measurements: mechanic arm moves inside the bin to do its grasp job strategies, placing or isolation, all fields are based on the given camera position and point cloud found out has the same problem but its confrontation on information provided.

To guarantee a better freedom to the motion planner algorithm we implemented a saturation block that cut off unfeasible movements for the robot and ensures that robot avoid the shelf, in particular camera uncertain often are saturated, these safety net permit us to remove possible damaging for the environment and users which work near the structure.

Nevertheless the advantages showed, the technique is not perfect, in fact the off-line calibration is needed and, also if it is bit long, because robot is moved on the main directions toward an the environmental constraint, for each bin.

To maximize reachable positions we choose to set as limits the first points where a substantial deformation was available.

Although configuration requires a long time it can be partially automated, robot reach a bin and thanks to the preset deformation measurement we understand when a wall or a ceiling is reached and we reduce the find out coefficients to increase robust properties.

A test phase start after that calibration is end, robot retrace all saturated position or further add are necessary, one classic example is when the shelf needs particular modification to avoid the wedging of robot and shelf, during test we are able to modify saturation to use in the code.

Values depends on several properties: number of the bin, different depends from position but, at the same time, the distance from bin is settable, in fact the mouth of each bin is smaller of entire internal dimension.

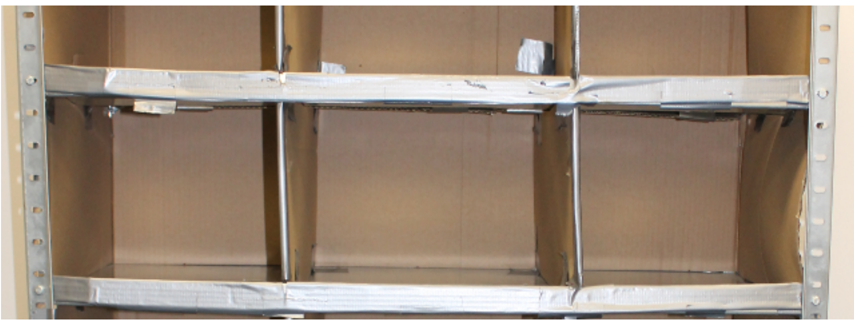


Figure 6.6: Gate dimensions.

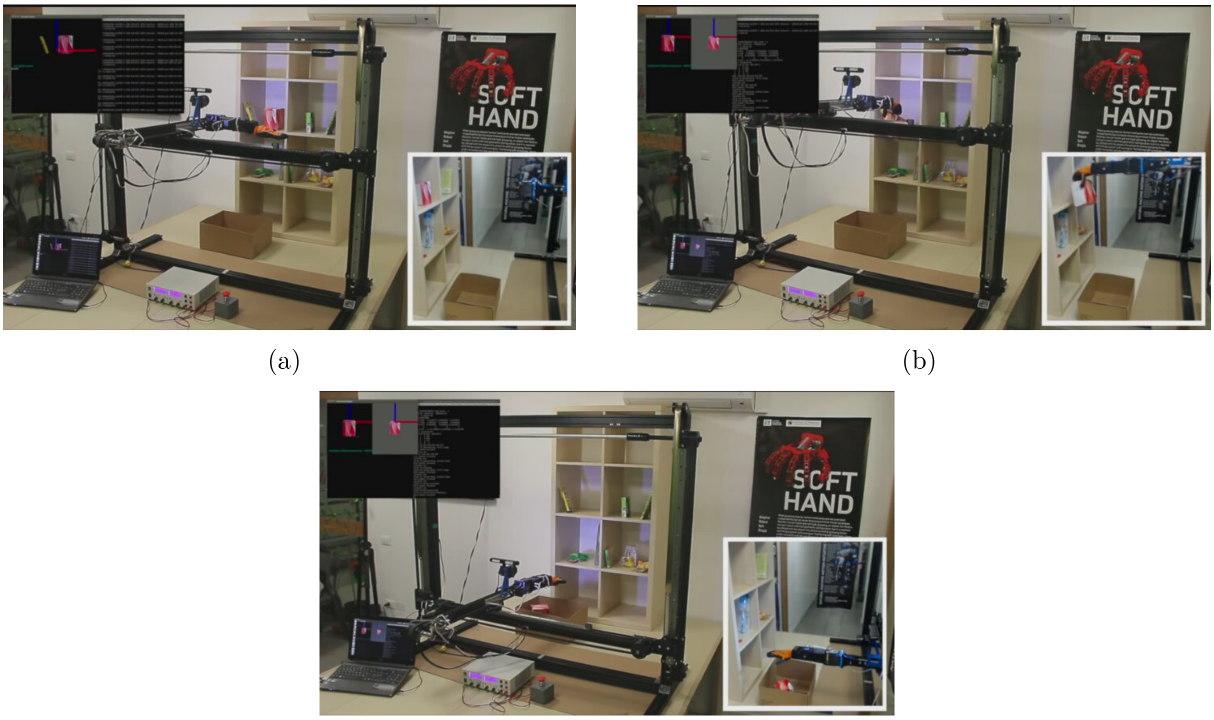
Automated part is based on the deflection measurements imposed to motors, which are subjected to the extension, when motor shaft reach an external obstacle the deflection increase and robot stop its motion.

### 6.3.3 Amazon Picking Challenge

During APC the robot operation had to be fully automated and the human operator was never allowed to interact with the system during all the process. The JSON file contained a list of 12 objects and the robot was supposed to execute until all the stuff was taken, or until the team leader decided for it to stop.

Our team was selected to receive the travel support prize thanks to a video that presents our work, it was not the final result but a preview of the intentions and presents the work done until that moment.

In Fig.6.7(a)-6.7(c) a photo sequence of an example task is shown, JSON file contains only three elements but the skeleton of the architecture keeps the same.



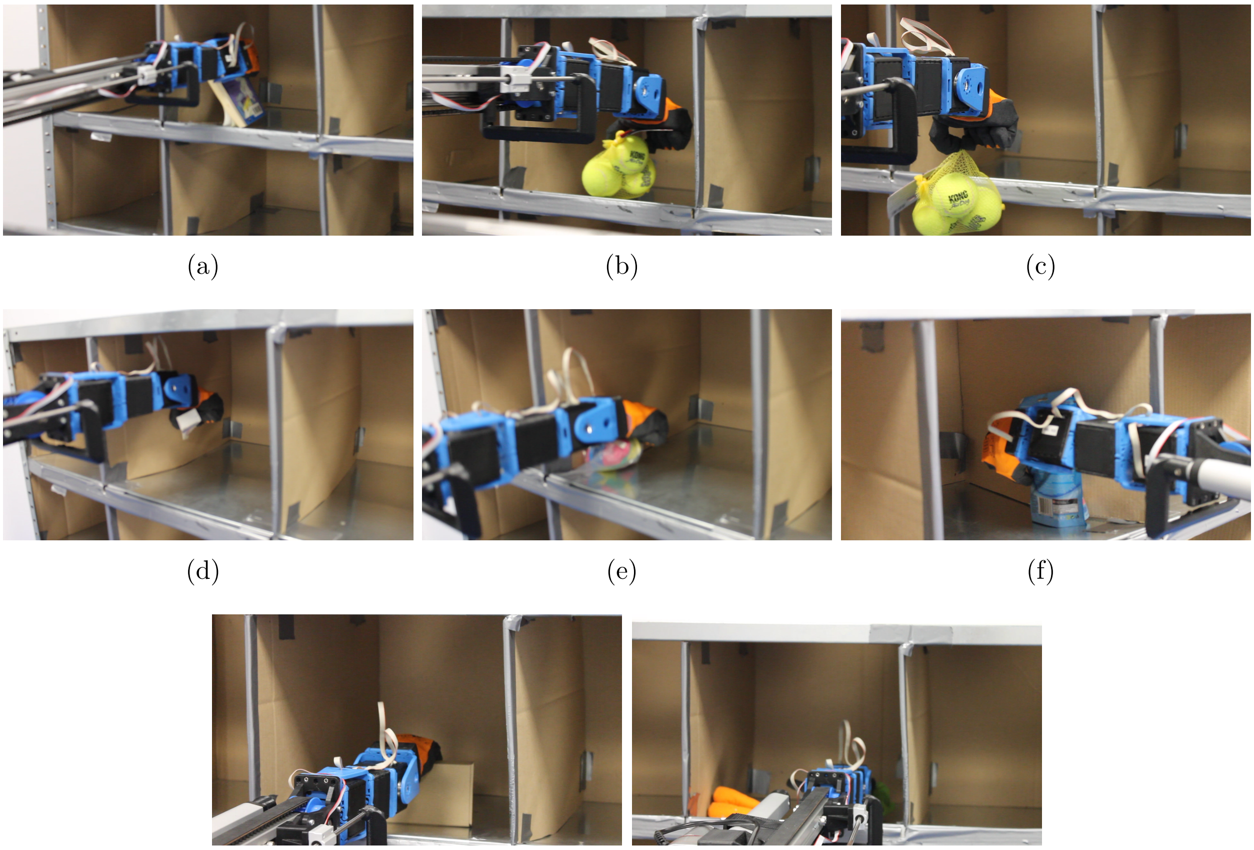

(a) (b) (c)

Figure 6.7: Photo Sequence of the task.

Fig. 6.8 shows robot configuration during the real challenge.

Figure 6.8: Robot during the challenge

## 6.4 Further Experiments



(a) (b) (c) (d) (e) (f) (g) (h)

Figure 6.9: Fetching objects.

# Chapter 7

# Conclusion

During this thesis, we investigated an interesting approach, proposed in recent years, based on SoftRobot which treat environmental constrains (EC) to perform better grasp.

In the first stage of our work we made experiments on grasp in constrained environment, the chosen working framework was a shelf, which represent a classic work situation for an human been in a lot of factory. We chose a collection of items which was representative of a more general set of objects, we covered several dimensions, shapes, softness and deformations properties.

We made experiments through an Handle, a device which permit to use Pisa/IIT Soft-Hand as substitute to human hand, by a lever we control closure of the gripper. Thanks to this tool a tester tried to grasp a set of objects posed in a shelf, for each items we tested all admissible positions and orientations and we gathered informations for further analysis.

After experiments stage we studied data and we proposed a classification to discriminate all grasp strategies observed. These groups are very general and in some cases they are comparable one to another, due to this observation we made further sub-classification to understand better which parts are essential and which are derived.

Last part of analysis consisted in further statics about the set of items tested and results come from experiments, we observed that the most probable grasp strategies depends on which zone of shelf we are evaluating: in center zone is CP (Center and Push down), in left zone is MS (Hand 90° and Push on the Left) and in right zone is MTD (Hand $-90°$ and Push on the Right).

These three grasp strategies are the easier in their zone and we expected this result thanks to high flexible properties of Pisa/IIT SoftHand which keeps simple most of grasp situation because of its adaptability.

In the second stage we implemented results on a physical structure which is composed by three classic linear motors, three qbmove (VSA), an electric piston and Pisa/IIT SoftHand.

Introducing VSA motors on our robot we need a different kind of motion planner which allow us to exploit stiffness modulation and EC in the scene without problems, we decided to implement our own motion planner which permit all function that we need but, at the same time, it made a simple collision avoidance on known environment.

Taking informations from an external perception node and knowing the object to be picked the motion planner algorithm decide which is the best grasp strategies to execute it.

When a grasp strategies is triggered, depending on situations, several functions start to gather all essential informations, for example position to be reached (object center or object corner).

We managed multi-items events using softness and robustness of Pisa/IIT SoftHand, in fact when the target is surrounded, the items near desired object are pushed away to permit a clean grasp.

During motion planner logic part a number of vector states are stored and when a section of the code end we execute all saved configurations, each state has a collection of informations: robot configuration, preset values, rate value, closure rate, and etc.; we generate a trajectory which connect all these states following the request parameters, we imposed that all actuators reach the desired position at the same time.

We repeat experiments through the structure and we presented results produced by this approach.

Next stage will be to generalize approach on all kind of items and make further investigation on grasp strategies to sharpen classification.

# List of Figures

# Bibliography


[All11] M. G. Catalano at All. “SA—CubeBot. A modular variable stiffness platform for multi degrees of freedom systems”. In: *Proc. IEEE Int. Conf. Robot. Autom.* (2011).

[All14] M. G. Catalano at All. “Adaptive synergies for the design and control of the Pisa/IIT SoftHand”. In: *Int. J. Robotics Research* (2014).

[All15] R. Deimel at All. “Exploitation of environmental constraints in human and robotic grasping”. In: *Int. Symposium on Robotics Research (ISRR)* (2015).

[Aur91] Franz Aurenhammer. “Voronoi Diagrams - A Survey of a Fundamental Geometric Data Structure.” In: *ACM Computing Surveys* (1991).

[Bes92] N.D. McKay Besl Paul J. “A Method for Registration of 3-D Shapes”. In: *IEEE Trans. on Pattern Analysis and Machine Intelligence* (1992).

[Bic91] A. Bicchi. “Analysis and Control of Power Grasping”. In: *IEEE International Workshop on Intelligent Robots and Systems* (1991).

[BK00] A. Bicchi and Vijay Kumar. “Robotic grasping and contact: A review”. In: *Proc. IEEE Conf. on Robotics and Automation* (2000).

[Epp15] Brock Eppner. “Planning Grasp Strategies That Exploit Environmental Constraints”. In: *Int. Conf. on Robotics and Automation (ICRA)* (2015).

[LaV98] Steven M. LaValle. “Rapidly-exploring random trees: A new tool for path planning”. In: *Technical Report (Computer Science Department, Iowa State University)* (1998).

# Ringraziamenti

Il primo ringraziamento va a mia Madre e a mio Fratello che mi hanno appoggiato durante tutto il mio percorso di studi (e non), senza mai vacillare, rappresentando una sicurezza a cui poter far riferimento in ogni momento.

Durante il mio corso studentesco ho trovato tanti amici ed ognuno, a suo modo, mi ha dato una mano per raggiungere la meta ma un posto particolare nei ringraziamenti lo riservo per Emanuele, Mattia, Lorenzo e Alessandro con cui ho condiviso nottate di lavoro, studio e *amarezze*, sperando di aver imparato qualcosa da ognuno di loro.

In fine un ringraziamento va anche a tutti quelli che si sono adoperati per darmi una mano, Sabato o Domenica che fosse, durante la tesi o progetti di varia natura, i quali mi hanno dato fiducia che spero di riuscire a contraccambiare.

Sono stati lunghi anni che mi hanno insegnato tante cose e regalato tante emozioni: ansie, delusioni, felicità, gioie, arrabbiature; ma ogni volta che avevo bisogno di condividere queste esperienze sapevo sempre da chi andare, lei avrebbe lenito la delusione o partecipato alla mia felicità come fosse la sua, credendo in me, forse troppo, ma senza remore, grazie Alessia.