Final Master Thesis

Master in Automatic Control and Robotics

Robotic manipulation skills for picking and unfolding garments

Author: Irene Garcia Camacho

Director: Júlia Borràs

Ponent: Cecilio Angulo

Call: September 2019

Universitat Politècnica de Catalunya

Escola Tècnica Superior d'Enginyeria Industrial de Barcelona

Institut de Robòtica i Informàtica Industrial

Abstract

Robotic manipulation of textiles is a very hard task that is receiving lot of attention from the vision and robotics research community. It is hard because perception of textiles should account for many different configurations and varying shapes, and manipulation differs a lot from classical object manipulation in precision and ability. This Master Thesis simplifies the perception part to concentrate in the manipulation problem. Real experiments are performed using a TIAGo robot and custom made grippers. We first realize that manipulation skills require a rather high precision in the positioning, so the hand-eye calibration of the robot is evaluated in the task of grasping a folded cloth. Second, several baseline skills (simple grasps and motions in open loop) are developed and combined to construct two different complex manipulations: spreading a tablecloth and folding a towel. Finally, the manipulations are evaluated using a new proposed benchmark.

Contents

List of Figures

List of Tables

List of Acronyms

IRI Institut de Robòtica i Informàtica Industrial

CLOTHILDE Cloth manipulation learning from demonstration

HuMoUR Markless 3D human motion understanding for adaptive robot behavior

DOF Degrees Of Freedom

FSM Finite-State Machine

HRI Human-Robot Interaction

RGB-D camera camera that gives normal RGB image and its corresponding depth image

ROS Robot Operating System

TCP Tool Center Point

DMP Dynamic Movement Primitives

1 Introduction

In the past years, robotics has been gaining popularity, particulary in social and assistance domains with the aim of helping humans in their daily life activities as well as empowering elderly people or with reduced mobility. For this reason, the robotics research community has as one of the main goals to improve robot's abilities and performance in order to interact with people and help in human environments, making them safe and adaptable to non-predefined environments.

People's daily self-care activities are grouped onto what is known as Activities of daily living (ADL), activities for basic functioning, and Instrumental activities of daily living (IADL), which are not fundamental activities but let individuals live independently in a community [1]. Expectations on the capabilities of robots are very high [2]. One of the main tasks an assistant robot should perform is related to the manipulation of objects. Until now, efforts have been normally focused on handling rigid objects. However, we found that assistant robots will also have to deal with non-rigid objects, like textiles, as they are commonly present in the human environments. Cloth manipulation is necessary in houseworks such as doing the laundry, setting a table or making a bed or for dressing a person with reduced mobility. Given the flexible and deformable nature of cloths, its manipulation entails new challenges in the perception techniques and in the robotic manipulation itself. Therefore, it is required to develop new strategies to provide the robot with the necessary skills to autonomously fulfill basic functions as picking or handling, folding or unfolding fabrics. For example, the ability of perceiving and localizing a determinate point of a garment, such as a corner or an edge, and perform a fine grasping of it with the robotic arm is a key factor towards the accomplishment of more complex tasks.

In this master thesis we start studying the state of the art in cloth manipulation, defining the most relevant tasks and studying the strategies currently used in the literature (Section 3). We also identify the new challenges that appear in comparison with rigid-object manipulation and define the involved skills necessary to solving them. In Section 4 we focus on a basic feature measuring the precision of the hand-eye calibration, which is used in the problem of bringing a robotic arm to a determinate position obtained with a camera. Finally, two baseline solutions are proposed, for which performance is evaluated (Section 5).

1.1 Objectives

When we manipulate garments with the objective of performing some action with it, we usually do it by grasping the cloth from key points, which can be generic, such as corners or edges, or specific to the type of cloth, such as shirt collars or the waist of a trouser. To do so, it is necessary to make use of a robot which has the ability to move the arm precisely to that specific point. The main objective of this Master thesis is to assess the capabilities of the robot TIAGo in manipulating deformable objects. The present work wants to give answer to three scientific questions:

- Can we dotate a robot the ability to autonomously manipulate a garment?

As a starting point, we define some relevant tasks involving garment manipulation, studying the state of the art of the strategies adopted. According to these tasks, manipulation skills that allow the robot to execute autonomously textile manipulation tasks in an environment with uncertainty are identified.

- Does the hand-eye calibration give the sufficient precision to be able to grasp a unique layer of a folded garment?

  We want to asses the validity of the hand-eye calibration to perform fine manipulation in a variable environment without the need of extra sensory monitoring, only using a RGB-D camera and TIAGo's robotic arm. To do so, a basic task where the robot autonomously localizes and grasps a corner of a folded garment placed over a table is implemented . The validity of this calibration will be given with the success in the result of this task without human intervention. The precision will also be measured with a high pose accuracy system (HTC Vive system), obtaining the error between the selected grasping point obtained through the camera and the final position of the robotic arm.

- Is a basic strategy enough for performing more complex tasks relating cloth manipulation?

  Baseline solutions are presented, following a naive strategy for solving two tasks such as spreading a tablecloth and folding a towel. These tasks will help in assessing if a basic strategy is enough to continue working with deformable objects or more complex skills and ad hoc solutions are necessary to perform autonomously assisstance tasks. This tasks will be evaluated with performance metrics based on the success of the task, the execution time and the quality of the results based on a new benchmark for dual-arm cloth manipulation.

1.2 Motivation

This thesis is under the scope of the european project CLOTHILDE[1] and the national project HuMoUR[2]. The CLOTHILDE project (cloth manipulation learning from demonstration), aim to establish the foundations of versatile cloth manipulation by robots. Its aim is to develop a theory of cloth deformation under manipulation leading to a general framework for robots to learn to manipulate garments from human demonstrations. Such framework will encompass among other things robot perception and skill learning. On the other hand, the goal of the national project HuMoUR (markless 3D human motion understanding for adaptive robot behaviour) is to develop novel computer vision tools to estimate and understand human motion using a simple camera and use this information as a demonstration to teach general purpose robotic assistant to perform new complex manipulation tasks (refered as learning from demonstration).

[1] https://clothilde.iri.upc.edu
[2] https://www.iri.upc.edu/project/show/193

The present project wants to provide a baseline to define if extra hardware or other resources, such as visual servoing, sensory feedback, etc, are necessary to be included to the system to continue with manipulation of deformable objects. As the previous projects develop complex vision algorithms, the focus of this work will center on the manipulation skills, implementing a simple vision algorithm in order to work, but considering that they will make use of more advanced perception skills.

2 Resources

This project has been conducted at the Institute of Robotics and Industrial Informatics (IRI).

This section presents the resources used to develop this project, both hardware and software. The most important hardware resource is the robot itself, which in this case are two TIAGo robots. These robots are used for developing the practical applications and experiments for obtaining the presented results. On the other hand, the most important software resource used are Robotic Operating System (ROS), C++ and MATLAB.

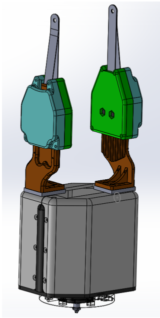
2.1 TIAGo robot

TIAGo (Take It And Go) is a mobile manipulator robot from the company PAL Robotics (Fig. 1). Its modular architecture allow multidisciplinary applications such as navigation, perception and manipulation, making it an ideal robot for research activities.

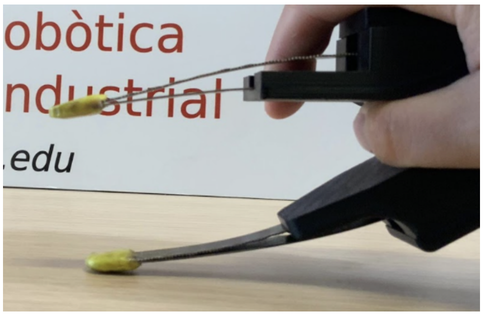
It provides workspace versatility with its 12 degrees of freedom, without the end-effector, 7 of them corresponding to the arm, which can carry up to 12Kg of payload. As end-effector it can either have a five-finger hand or a parallel gripper and it also has a force/torque sensor in its wrist. On its base, it has a laser range-finder which allow an autonomous navigation with obstacle avoidance and an emergency button, necessary as it will be destined to human collaboration environments. Besides, it incorporates an RGB-D camera mounted on its head used for perception applications and a stereo microphone and a speaker useful for human-robot interaction purposes.

Concerning the software, it is 100% programmed with ROS.

Figure 1: TIAGo robot.



(a) CAD of the modified gripper.



(b) Flexibility of modified gripper.

Figure 2: Modified gripper for TIAGo.

2.2 End-effector

At the Institut de Robòtica i Informàtica Industrial (IRI), a gripper which satisfies the specifications to carry out the tasks described in this document has been designed. The principal characteristics of this gripper, seen in Fig. 2, are its fine and flat structure, what makes it ideal for performing fine manipulation (e.g. enter between two layers of fabric), and its flexibility, in order to collide with the table without causing damage neither to the robotic arm nor to the table. This fingers are used with a 3D design part which is attached to TIAGo's parallel grippers (Fig. 2a).

The material of the fingers also allow to slide without too much friction in order to be able to perform edge tracing of a garment. Nevertheless, the finger tip provide enough strength in order to pick a garment without slipping.

2.3 Robot Operating System

ROS (Robot Operating System) is an interface that provides libraries and tools to facilitate the development of complex applications for robots. ROS provides the standard services of an operating system such as hardware abstraction, device control at low level, implementation of shared functionalities, message transmission between processes as well as data packages management.

Its operation is based on the communication between system modules, called nodes or packages, by exchanging messages, which can be of three kinds:

- Topics: used in bidirectional communication channels between two different nodes, where the transmitting node, called Publisher, sends periodically information regardless of the existing receiving nodes (Subscribers). It is the simplest kind of communication between nodes since there is no answer from subscriber and therefore no way of knowing whether the information has been received correctly.

- Services: the exchange of messages between nodes running in this mode work as a server/client communication. That is, data is only published when a request is made by a client, suspending its process until getting the answer from the server.

- Actions: they are based on the same principle as the previous bullet.

2.3.1 Modules architecture

The use of ROS services and actions means having to handle a multitude of variables, functions and callbacks for each action or service. In addition, for each action and service, two state machines are linked (for the client and the server) in order to follow their status and manage the transitions of states between them during execution. For this reason, in applications like those implemented in this project in which a multitude of actions, services and topics will be necessary, the complexity in the management of errors increases.

A new modular based structure has been implemented at IRI, which is used for the development of this project. It consists on creating a module for each functionality or operation of the robot (navigation, play motion, control gripper, move arm, etc.), which can be used as libraries in other applications using a simple API. The modules are arranged in a hierarchy tree class. Each module implements a state machine responsible of managing the ROS interfaces of the actions and services involved, so that the use of the robot's functionalities is reduced to the access through the API provided by each module, reducing the number of ROS interfaces used directly.

The modules implemented are the following:

- *Arm_module*: Executes planned trajectories to move the arm to a desired cartesian position or to determinate joints positions.

- *Gripper_module*: Opens and closes the gripper. It also allows to move the fingers independently.

- *Head_module*: Moves the pan and tilt of the head to determinate angles or to look to a cartesian point of the space.

- *Move_platform_module*: Used to move forward, backwards or rotate the platform some determinated meters or radiants.

- *Nav_module*: Provides an autonomous navigation to move the robot in any place of a map.

- *Play_motion_module*: Executes pre-recorded motions on Tiago with its arm and torso.

- *Torso_module*: Used to move up or down the torso.

Besides, some other advanced modules, which implement other functionalities of the robot using the previous cited modules are:

- *Following_module*: Implements an application for following a person.
- *Guiding_module*: Module used for guiding a person to a given place.
- *Grasp_module*: Executes a pipeline for grasping a given rigid object in a determinate place.
- *Head_search_module*: Used for scanning the environment in search of QR markers, persons, etc.

The present project will make use of most of the basic modules, in concrete the ones for controlling the arm, gripper, head, torso, play motion and move platform. It is remarkable that the grasp module can not be used for this project, as it is an ad hoc solution designed to perform grasps of non-deformable objects. This implies that it does not take into account several aspects involved in manipulation of deformable objects, which will be described in Section 3.

2.3.2 ROS interface for HTC Vive

IRI has a virtual reality platform (HTC vive) which can be used for motion capture due to its high pose accuracy of its trackers and controllers. For this platform, a ROS package was developed at IRI which transfers the SteamVR poses into a ROS TF framework[3]. It is used for calibrating a robotic arm into SteamVR system, having both TF frameworks (one corresponding to the robot and the other one of the HTC system) in a same TF tree. For instance, with this system it can be tracked with a robotic arm one of the HTC controlers.

This resource will be used to obtain the results of the precision of the grasping of TIAGo after performing a hand-eye calibration, obtaining the distance between a HTC tracker attached to the garment and another one mounted over the Tool Center Point (TCP). As this ROS package was implemented for another robot (WAM robot), during this project it is adapted for its use with TIAGo robot.

(a) HTC Vive controller.

(b) HTC Vive tracker.

(c) HTC Vive base stations.

Figure 3: HTC Vive devices.

[3]http://wiki.ros.org/tf

3 State of the art of garment manipulation

When developing assistance robots for helping people in human-centered environments, it is required to identify the entities with which the robot will have to work. In addition to the common rigid objects, there are also many deformable ones like plants, cloths, food or even animated beings as pets or the people themselves. In regards to robotized manipulation, the process of handling rigid objects has been extensively studied, considering it a mature field in robotics community, as opposed to the study of deformable materials, in which the non-constant and variable shape of the objects invalidates the previous perception and manipulation techniques used for solid items, as the complexity grows when it comes to deformable objects manipulation.

This section presents the main challenges and some of the new considerations that have to be taken into account when dealing with deformable objects manipulation. It is also defined the most relevant tasks concerning garment manipulation, studying the necessary skills involved as well as the solutions given in the literature. It concludes proposing two significant tasks that includes several of the manipulation challenges, for which baseline solutions are implemented in Section 5.

3.1 Main challenges

Moving form manipulation of rigid objects to deformable objects implies several new considerations to take into account. In first place, the principal challenge presented is due to the difference in the representation of the state of a rigid object and a deformable object. The state of a rigid object is determined by 6 variables for position and orientation. This does not apply to non-rigid objects since a manipulation on them produces a deformation in its state, having an incomplete representation with only this 6 variables. This fact implies that in contrast with rigid objects, which state is invariable, textile manipulation requires to address its deformation caused after it has been grasped in order to perform the following action.

In second place, and as consequence of the previous problem, the current vision techniques used for rigid objects can not be applied to deformables objects. This fact makes it a field of interest to start with before focusing on the necessary manipulation skills since it is desirable to develop vision algorithms to classify the type of garment, identify its state and locate key points to effectively grasp it. For this reason, most of the works dealing with cloth manipulation focus on the vision algorithms rather than in the manipulation skills themselves.

In third place, in terms of manipulation skills, until now it was not necessarily required to perform fine manipulation when grasping a rigid object, being enough a coarse positioning of the gripper using compliant and soft hands which adapts to the objects. For example, for grasping a glass or a can it is not required to identify a grasping point being possible to handle the solid item in any way or at any place using a rough gripper. In contrary, this is not the case in textile manipulation since for their proper handling the objects should be grasped from a specific point (e.g.

collar in T-shirts, corners in fabrics, trousers, etc). For this reason, it is necessary that the robot has the ability to move the arm to a precise position sensed through a camera.

Only recently rigid object manipulation studies have started to consider the possibility of using the environment to manipulate, for instance, by first touching the table where the object is. While this is optional for rigid objects, it is usually necessary in garment manipulation since the nature of textiles makes them adapt to their environment. This entails the design of new specific grippers and strategies which takes into account collisions without damaging neither the robot nor the environment objects.

As it has been said before, the first challenges related to the perception of textiles are not in the scope of this work as they are treated in the projects presented in Section 1.2. Nevertheless, in the context of this Master thesis is to study common manipulation skills for giving an idea about whether it is possible to have general solutions for solving autonomously complex cloth manipulation tasks.

3.2 Relevant tasks

In recent years, interest in the development of robotized tasks for cloth manipulation has increased. However, as it is a novel field a generalization in the methods to perform the perception, the manipulation, or the gripper design cannot be found. Instead, there are many different approaches to solve similar tasks and even more types of gripper designs for each type of task. A good recent review of the grippers that have been used so far for clothing handling can be found in [3]. The mentioned challenges are some of the reasons why we found these discrepancies in literature when handling deformable objects with robots. Nevertheless, similarities in the basic manipulations between solutions for solving a same task can be identified. For this reason we review the current state of the art classifying them according to the type of task, as done in [3].

3.2.1 Pick & place

This type of task is very common when starting to develop textile manipulation applications, since it is the most basic function. It is performed for instance for separating a garment from a pile, for hanging a piece of cloth or for cloth identification through 3D reconstruction. Regarding crumpled garments its principal challenge comes to perception in order to identify a garment from another one [4, 5, 6].

3.2.2 Unfolding in the air

This task consists on the action of grasping a garment from a crumpled or folded state and manipulate it to get it extended and grasped by two relevant points. It is usually performed as a preparatory action for performing the next manipulation, for instance grasp a T-shirt by its shoulders for its classification or grasp a tablecloth from two corners for spreading or folding. As it can be deduced, it is a bi-manual task which needs of two robotic arms in order to execute it. To perform this task it is necessary to provide a robot with vision skills for localizing the suitable grasping points for having it unfolded in the appropriate way. In addition, manipulation skills are also required in order to grasp correctly the garment from the identified points for bringing the cloth into the desired configuration. This manipulation requires basically the ability of performing precise grasps, i.e. move the end-effector to a determinate location with small distance error, what would reduce the task execution time by reducing the number of re-graspings of the fabric. The design of the gripper will significantly help in the performance of this task, meaning that a rough gripper with no precision in the grasp will be inefficient.

So far there have been different strategies to achieve this task. The first one consisted in grasping the lowest hanging corners of a cloth that is grasped by one hand, or use a perception method to identify the grasping points. Repeating the task alternating between hands, then the vision system had to identify when the cloth was already grasped correctly [7, 8, 9, 10]. Another approach was to slide the grasped cloth over a table to make the grasping points more visible [11]. One of the newest approaches grasps only the first corner of a rectangular cloth, and then to get to grasp the second one it grasps a point on the edge and slides the hand until it reaches the other corner [12], in an operation that will be called *edge tracing*.

Depending on the strategy used, the manipulation skills complexity increases. For example, execute an edge tracing with the second gripper until reaching the second corner as humans do has more complexity but it decreases the need to localizing the second grasping point.

3.2.3 Spreading or flattening

Once the garment is unfolded in the air, that is, flat and grasped by the correct grasping points, it can be placed flat on a table. To do so, there are also several strategies. Moving the the cloth slowly over the table using the edge of the table to make sure it is placed flat is the most common one [8, 9, 10]. Another strategy involves complex dynamics in order to achieve the desired behaviour of the garment to spread it [13]. If a flat cloth has wrinkles or undesired folds, then it has to be flattened. Strategies to flatten a cloth must have into account contacts and friction with the environment. This last task is commonly studied in order to start testing approaches of manipulation where slight collisions are present. Besides this considerations in the manipulation, perception algorithms are necessary for detecting wrinkles and creases and to identify which type of manipulation is necessary in order to remove them [14, 15, 16].

3.2.4 Folding

The task of folding a garment is a very popular task since it has a great potential of being used in industrial applications and in domestic environments. This task presents a great variety of solutions depending on the type of garment to fold, since the process for folding a T-shirt, trousers or a rectangular fabric is very different. Concerning the most basic garment; a rectangular garment like a towel the process also differs depending on the gripper used and the technique. Concerning bi-manual strategies, examples of two solutions can be given; fold on table or folding in the air. The first method consists on grasping two corners of a spread fabric and executing a coordinate arm motion with both robotics manipulators until attaching two and two vertices of the garment. This is the most used approach [8, 17, 7, 18, 19]. As it can be observed, this task needs vision skills for recognizing where the folding lines should go [20], and manipulation skills for compute the manipulator trajectories [21]. In addition, other some sensory feedback can be used for monitoring the cloth state after the fold. On the other hand, folding a T-shirt or a shirt in the air is a complex manipulation consisting in grasping the shirt by the shoulders and performing the fold in the air by rotating the grippers, then placing the semi-folded shirt on the table and perform the last fold before releasing it folded on the table. This is attempted in [22] and [9].

3.2.5 Dressing

Tasks focused on dressing (putting a T-shirt, jacket, shows or scarf) are used as assistance applications for elderly or mobility reduced people. This is the tasks which entails higher complexity as an important entity comes into play; humans. This implies a very important challenge not yet mentioned that is the safety in the performance of the task, meaning that new considerations must be taken into account as for example the robot's compliance, prediction of the future motion of the person, etc.

3.3 Proposed tasks

We propose two textile manipulation tasks; spreading a tablecloth and folding a towel, which combine several of the sub-tasks mentioned. We list hereunder the involved sub-tasks and the manipulations required to perform them:

1. **Spreading a tablecloth**
    - Unfold in the air the tablecloth: Grasp the tablecloth by two corners to have it extended.
        - *Grasping:* Grasp a corner of the tablecloth from two possible starting configurations: either folded or crumpled on the table.
        - *Edge tracing:* Grasp a point on the edge and trace the edge until the second corner is reached.

  - *Spreading*: Spread the tablecloth over the table.

2. **Folding a towel that is flat on a table:**

  - *Grasping:* Grasp the corners of the cloth.
  - *Folding:* Perform the folding action and release.

The first task is novel for several reasons. First, it has not been done before with robots, secondly, manipulation of big clothes such as bed sheets or tablecloths is also novel and challenging. Finally, unfolding a cloth that is already folded has also not been done before. Therefore, putting a tablecloth is a good example task because it involves most of the challenges of cloth manipulation, while still it is a novel solution that includes new challenges.

In addition, we have chosen the folding task because it is one of the most common tasks implemented and therefore, it is a good task for comparison purposes. To solve it, it needs very similar skills than the previous task and therefore, it will be used to exemplify how the implemented skills can be used for other common tasks.

We can identify several challenges, which need different type of manipulation skills to be solved. Here we list some of the manipulation challenges and the pipeline to solve them:

- Grasp a corner that lies on the table:

  1. Move the robotic arm to a determinate point of the garment sensed with the camera, such as the corner.
  2. Collide with the table where it is placed.
  3. Perform a clamp of the fabric.

- Grasp a corner of a folded cloth:

  1. Place the bottom finger of the gripper between the folds with precision.
  2. Perform a clamp of the top layer of the folded garment.

- Perform edge tracing:

  1. Grasp a point on the edge of the tablecloth.
  2. Execute a complex trajectory with the robotic arm for sliding along its edge.

- Spread the tablecloth:

  1. Move the grasped tablecloth over the table until covering it.

- Perform the folding:

  1. Compute the folding trajectories.
  2. Execute a coordinate motion of both robotic arms.

The practical implementation of these two tasks will help us to identify the strengths and weaknesses of two basic methods in manipulation; **manipulation through hand-eye**, that is to say, grasp a point which has been previously located by a sensor, and **manipulation through trajectories learned by demonstration**, the method that will be used to obtain the needed trajectories. It will also allow us to determine if it is necessary to use some type of monitoring in order to successfully perform these type of manipulations, as well as responding to whether it is enough to develop basic strategies and naive approaches to solve cloth manipulation tasks or if, on the contrary, smarter solutions are needed.

To implement these skills, we will use the resources from the Institut de Robòtica i Infomàtica Industrial mentioned in Section 2. In Section 5 we present baseline solutions for these tasks in order to test some of the identified manipulation skills.

4 Hand-eye calibration

In this section it is studied in depth a basic feature required by all manipulation skills: the ability of bringing with precision a robotic arm to a sensed point in the workspace. As seen in the previous section, the indispensable condition to perform any cloth-specific manipulation task (folding, unfolding, dressing, etc) in variable environments is that the robot can grasp with precision a determinate point of a garment that has been localized with a camera. This precision is classically achieved through a calibration of the robotic arm and the camera called hand-eye calibration [23].

4.1 Theoretical background

Hand-eye calibration is important in robotics systems for manipulation and grasping tasks. It first appeared with the problem of moving a robotic arm to a position in space which has been obtained through a sensor, normally a camera. To solve this problem it was necessary to know the relationship between the Tool Center Point (TCP) and the sensor. Therefore, the hand-eye problem wants to obtain the relative position of the robotic arm with respect the sensor, that is to say, compute the rigid transformation (rotation and translation) between a frame of the robot and the camera frame.

As it is mentioned in [24], the hand-eye calibration is important in two types of tasks:

1. *Move precisely a sensor mounted in a manipulator* (Fig. 4-left). In order to know how to move the robotic arm so the attached camera points to a determinate position, it is necessary to know which is the relationship between them. This is the case for example in visual servoing.

2. *Obtain the position of the sensed object with respect to a robot coordinate frame* (Fig. 4-right). This is the case when we want to grasp an object which position and orientation in space are obtained through a sensor. In order to be able to move the manipulator to its position, the transformation between the sensor and the Tool Center Point (TCP) must be known.

The project described in this thesis is related with the second task. The sensor, in our case a camera, is used to sense the object (a garment) in the environment. Once the position of the garment with respect to the camera frame is obtained, it is needed an accurate transformation to a known coordinate system of the robot, as can be the robot base frame, in order to know how to move the robotic arm. This calibration attempts to solve a homogeneous matrix equation of the form:

$$AX = XB \tag{1}$$

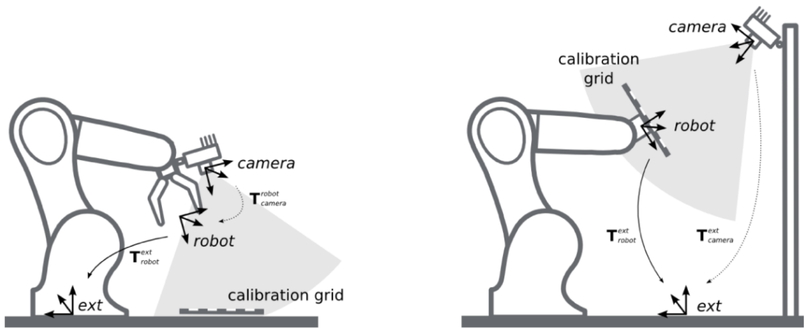


Figure 4: Handeye setups. Image source: https://doc.rc-visard.com/v1.3/en/handeye_calibration.html

where A is the transformation between two different positions of the sensor frame, B is the transformation between two positions of the TCP frame and X is the unknown transformation between the end-effector frame and the sensor frame. A, B and X are 4x4 matrices of the form:

$$A = \begin{pmatrix} R_A & t_A \\ 0 & 1 \end{pmatrix} \tag{2}$$

where R_A is the rotation expressed in a 3x3 orthogonal matrix and T_A is a vector describing the 3 coordinates of the translation.

Before obtaining the extrinsic parameters (rotation and translation of the X-transform), which are given as solution of the hand-eye calibration, it is first required to calibrate the intrinsic parameters of the camera, including the focal length, the optical center and the skew coefficient. Figure 4 shows the two setups for the hand-eye according to the type of task to perform, as it has been said, we will focus on the setup seen in Fig. 4 right. The hand-eye calibration process consists on bringing the robotic arm to several positions in space, making sure each of those positions are different from the rest so that the system obtain feasible and rich information. This positions will give the previous matrices A and B, considering not only 2 different positions but a minimum of 20 for solving the homogeneous matrix equation.

Nevertheless, the hand-eye calibration for the robot used in this project is performed with a small difference. In contrary to the previous cases where the sensor is placed at a fixed position or mounted on the robotic arm, the TIAGo robot has its camera mounted on its head over pan and tilt joints to increase its field of vision. In order to calibrate the robotic arm in all its workspace, during the calibration the head of the robot also moves in addition to the robotic arm in a way it always has in sight the calibration grid attached to the end-effector. The result is a wider calibrated workspace (showed with a gray shadow in Fig. 4) where to move the end-effector with precision.

It is worth to mention that MATLAB and ROS libraries are used in this project in order to solve the hand-eye problem and obtain the resulting fixed transformation between the camera and the robot.

4.2 HTC tracker

One of the objectives of this Master thesis is to study the precision of the hand-eye calibration for textile manipulation purposes. The goal is to determine if with this type of calibration we obtain enough precision to perform some of the tasks seen at Section 3, such as grasp with precision a corner of the fabric, without needing more complex solutions as for example visual servoing.

To do so, it is implemented a basic task consisting on bringing the Tool Center Point (TCP) of a robotic arm to a grasping point of a folded garment localized through the robot's RGB-D camera. This task will serve for obtaining precision results measuring the final distance between the corner and the fingertip. This could be done in several forms; measuring with a ruler the distance, which is not a very precise method. Obtaining the difference between the sensed point position with respect to (w.r.t) a fixed reference frame (e.g. world frame) and the fingertip frame as well w.r.t. the world frame once it has moved to the grasping point. However, this results do not give the desired real error between the final position of the end-effector and the garment, instead, it gives the error in the relationship between the coordinate frames of the robot (TF framework). That is to say, it is the robot itself that obtains and transforms the position of the sensed point to its own reference system, regardless of whether it corresponds to the position in the real world, and plans the motion of the arm to move it to that particular point. The best solution is to make use of a system, independent of the robot, which gives with high accuracy positions in the real world. This allows to compare two different points in space (the fingertip and the garment's corner) with precision. For this reason it is decided to use the HTC Vive system presented in Section 2.3.2 as it is recommended due to its high pose estimation accuracy.

As it has been said, the ROS interface of this system was developed for another robot available at IRI so it has to be adapted for this purpose for TIAGo robot. With this system we can obtain a single TF framework merging the TF tree corresponding to the robot and the TF tree of the HTC system, obtaining a relationship between the robot and the real world. With this purpose it is decided to use two of the HTC trackers, one mounted on the robotic arm and another one attached to the garment. Note that the resulting system consists on another hand-eye problem (having in this case the HTC controller of the arm as the sensor), since we want to know the exact position of the arm tracker with respect to the robotic arm. As TIAGo is a mobile robot, in order to merge both TF frameworks in a way that it allows the robot to change its position in the environment it is necessary to use a third HTC controller placed on the robot's back, resulting in a third hand-eye problem for obtaining the fixed transformation between the tracker and the torso's reference frame. Fig. 5 shows the resulting system with the two new hand-eye problems, marked in red the unknown transformations to compute and in green the transformation to obtain the error in the positioning of the gripper. The tracker's reference frames positions can be visualized in Fig. 6.

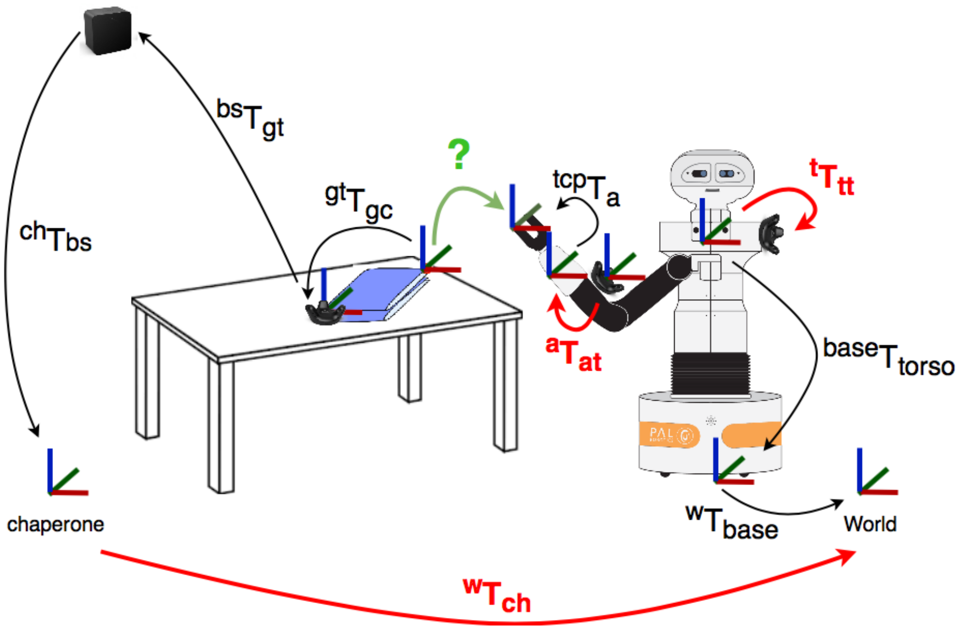


Figure 5: HTC Vive experimental system transforms.

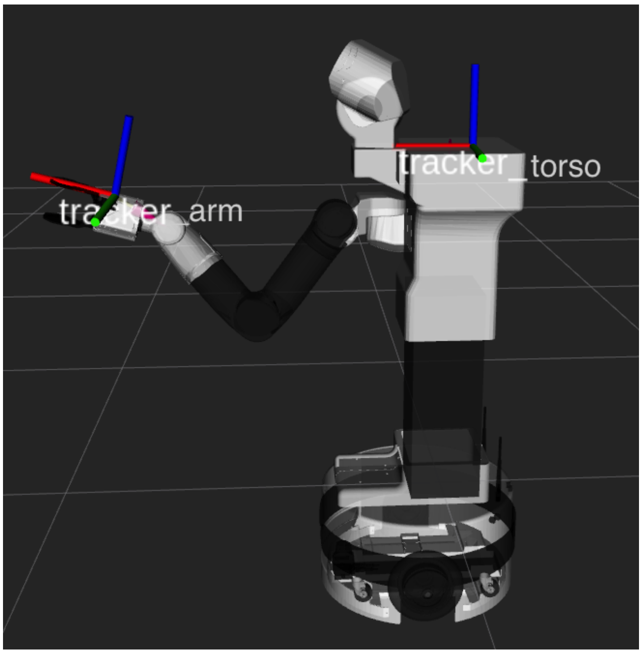
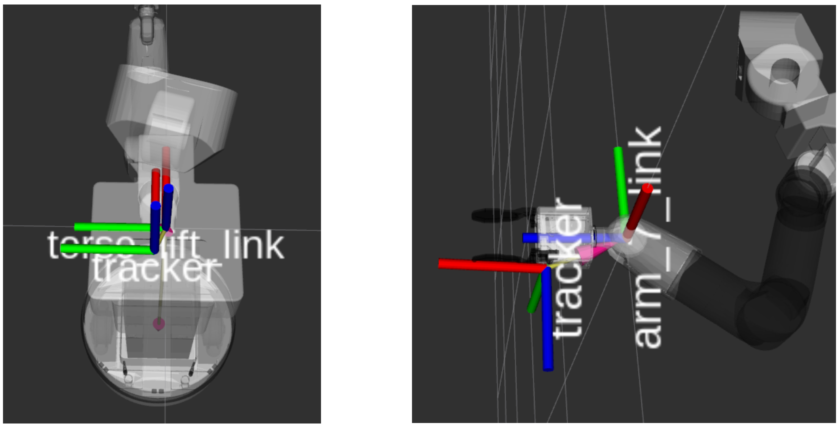


Figure 6: Visualization of the reference systems of the HTC tracker mounted over TIAGo.



(a) Torso tracker position.

(b) Gripper tracker position.

Figure 7: Tracker's reference system positions according to the hand-eye calibrations.

In order to correctly perform this calibration to obtain the new unknown transformations it is first necessary to fix both controllers to the robot so that they do not move during the experiments, what would invalidate the computed results. The calibration process consists on bringing the robotic arm to several positions of its workspace, in our case 50 positions, and saving the following transformations at each configuration:

- T_w^{at}: Transform from the tracker to the HTC Vive world reference frame, i.e. *arm_tracker* frame pose w.r.t. *chaperone* frame.
- T_t^a: Transform from the robotic arm to the robot base. In this case, as the robot base is mobile, we use the torso frame, i.e. position of *arm_tool_link* frame w.r.t. *torso_lift_link* frame.
- T_w^{tt}: Transform from the tracker to the HTC Vive world reference frame, i.e. *torso_tracker* frame pose w.r.t. *chaperone* frame.

With this transformation the two hand-eye problems can be solved, resulting in two fixed transformations (T_{at}^a and T_{tt}^t) corresponding to the position of the tracker attached to the end-effector w.r.t the robotic arm frame (*arm_7_link*) and the position of the torso's tracker w.r.t the torso frame (*torso_lift_link*), see Fig. 7).

Having this transforms, it is possible to merge both TF tree as mentioned before and therefore obtain the position of the TCP in the real world.

Going back to the goal of this section, we wanted to use this system to measure the precision of TIAGo in grasping a point of a garment obtained through its camera. In order to perform the experiments, a basic task is implemented. It consists on localizing a corner of a folded garment placed over a table and grasp it. This entails having the precision to grasp a single corner insterting the fingertip between several folded layers. The performance of this task will be demonstrated with its physicial success, that is if it has been grasped the correct layer or not, and with the quality on the precision given by the hand-eye calibration.

4.3 Perception algorithm

In a environment with uncertainty and variable initial conditions, to manipulate a garment it is necessary to develop a vision algorithm with which we can obtain the desired grasping point at any moment in any configuration.

As this project focuses on the manipulation part, the implementation of a complex vision algorithm which can classify the type of garment, its state, its wrinkles, etc is out of the scope. Nevertheless, it is necessary to develop an algorithm to identify with enough precision the grasping position, in our case the position of the corner of a garment. It should also perform this detection with the folded garment in any position of the work space, in order to obtain feasible solutions on the precision.

To do so, TIAGo's RGB-D camera is used, which is located at its head mounted over pan and tilt joints. In order to locate the garment several solutions could be implemented, such as through colour detection, Convolutional Neural Networks (CNN), plane segmentation, etc. Color detection would restrict the users to use a table and a fabric of a specific colour and a CNN would imply a training database and higher computational time. To avoid these restrictions, a segmentation algorithm is used to differentiate the garment from the table. From the input point cloud of the sensed environment, a plane segmentation is applied in order to obtain the biggest horizontal plane, corresponding to the table. The table and the objects on top of it are then classified into two different clusters accordingly to a threshold value. From now on, the point cloud corresponding to the table will be obviated since the cluster of interest is the one which includes the garment, but still considering the table as an obstacle for posterior collision avoidance.

The proposed grasping point is the nearest corner of the fabric with respect to the robot to simplify the arm's pose at the moment of grasping. Therefore, the cluster corresponding to the garment is subjected to filtering until the top closest point in the X-axis of the robot is located. Notice that the presented algorithm does not perform any corner or edge detection, so it needs the corner to be exposed to get a correct result, either with a folded or crumpled garment. However, in the case of a rectangular folded garment, this point will usually correspond to a corner, unless its edge is placed perfectly perpendicular to the X-axis of the robot. The obtained position of the garment's corner with respect to the robot's reference frame is then published through a ROS topic in order to give access to other modules such as the *arm_motion* module, presented at Section 2.3.1, in charge of the motion planning of the arm. In Fig. 8 it can be seen the garment's point cloud cluster sensed by the robot's camera and the obtained grasping point marked with a blue sphere.

4.4 Grasping strategies

Two grasping strategies have been implemented as proposal methods to validate the hand-eye calibration. Both consist on the same basis: a garment placed on top of a table, with which fine grasping must be tested by grabbing it's corner with precision. The difference lies in the grasping strategy; the first method consists on

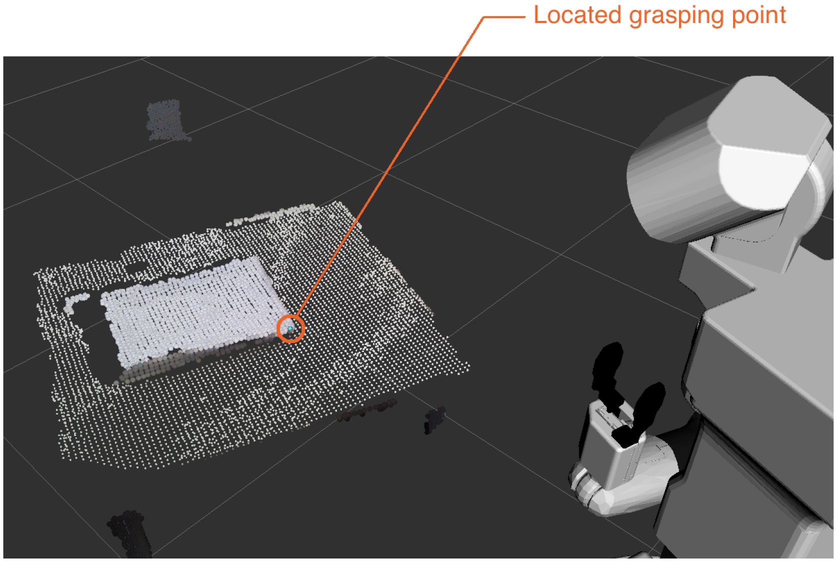


Figure 8: Visualization of the table and garment's point cloud and localized grasping point.

inserting the fingertip of the gripper between two layers of a folded garment, in order to grasp it by a single corner. In the second method, the pipeline to follow consists on picking the garment with a previous collision with the table, in order to slide the gripper in between the table and the fabric. The first strategy is needed in unfolding tasks, where a successful and feasible grasp by a single corner simplifies the following unfolding manipulation. On the other hand, the second strategy is useful for grasping a spread fabric over a table or can be used in tasks where a folded garment must be placed somewhere else without unfolding it.

The robot behaviour for both methods is controlled through a Finite-State Machine (FSM) consisting on a scan of the environment to localize obstacles around the robot, followed by the localization of the garment and its grasping point, and ending with the arm motion plannings to the grasping positions (pre-grasp, grasp, and post-grasp positions) in order to grasp the garment.

4.4.1 Relevant steps comprising the grasping tasks

- **Scan environment**

Before performing any motion with the robotic arm for grasping, it is convenient to generate a 3D representation of the environment which will be used for obstacle avoidance during the planning of the arm trajectory. To do so, a scan of the environment is performed by executing a sequence of pan and tilt positions with TIAGo's head that allows to fill a 3D voxelized representation of the environment that will eventually help to localize all the present obstacles, such as the table, with the robot's RGB-D camera. The information obtained with the depth sensor is used to update the collision matrix state, which defines the occupancy map of the environment. This action allows to execute safely movements with the arm through

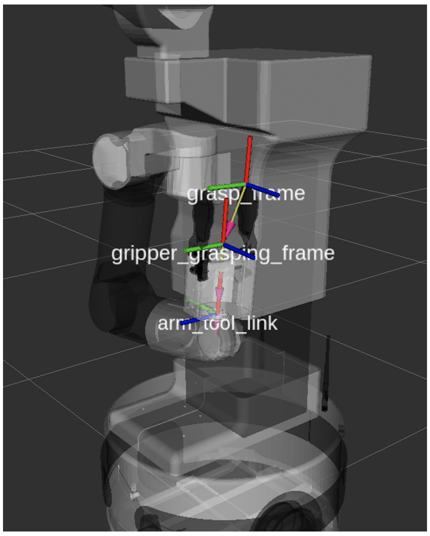


Figure 9: End-effector reference frames.

joint trajectory planning in order to reach a given pose in cartesian space considering collision avoidance with the objects.

- **Garment localization**

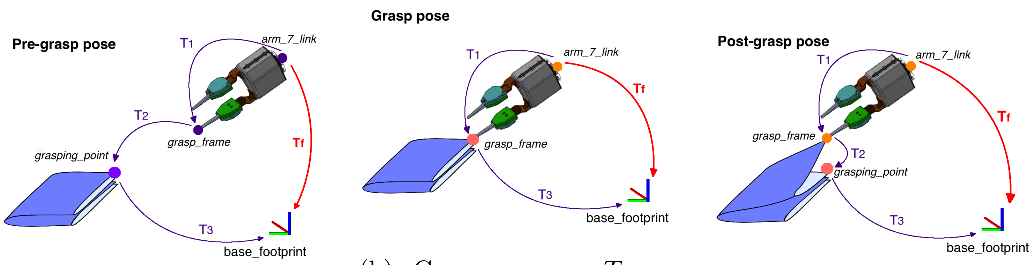
After the obstacles are localized, the robot must move its head so that it has in sight the table and the objects on top of it in order to obtain the grasping point of the garment as presented in the previous section.

- **Grasping positions**

This step includes several states of the FSM which are in charge of the computation of the arm positions for grasping the garment once the grasping position is obtained.

In order to facilitate the arm motion planning and execute short and clean trajectories during the grasping of the garment it is decided to divide it in three phases: pre-grasp, grasp and post-grasp. The pre-grasp phase will bring the robotic arm from the starting position to a suitable pose near the grasping point. The planned trajectory to perform this movement only needs to take into account obstacle avoidance in order to not collide with the table, performing a coarse positioning in terms of precision. In contrary, the following grasp position must be reached with a fine positioning in order to grasp the garment's corner precisely. The fact of bringing the end-effector to a near position of the grasping point reduces the distance of the planned trajectory, allowing to compute a shorter and linear trajectory, improving the precision of the movement between the pre-grasp and the grasp positions. As an example, setting the pre-grasp position at the same height of the corner but more distanced will facilitate the introduction of the fingertip below the corner of the fabric through an horizontal trajectory. Finally, the post-grasp position separates the gripper and the garment from the table to facilitate the next movement.

The arm motion planning of TIAGo robot works by computing a joint trajectory to bring the *arm_tool_link* reference frame, placed on the wrist of the arm, to a given



(a) Pregrasp pose.

(b) Grasp pose. $T_2 = 0$ Grasping position is the same as grasp pose.

(c) Postgrasp pose.

Figure 10: Representation of the frame transformations.

position and orientation in space. Notice in Fig. 9 that this reference frame does not correspond to the fingertip of the gripper, which is what is wanted to be moved to the grasping position. Therefore, a series of reference frame transformations must be computed in order to obtain the pose of the *arm_tool_link* necessary to have the fingertip at the given position. To do so, we start creating a new reference frame for the fingertip of the modified gripper, called *grasp_frame*. Fig. 10 shows the reference frames involved in the system:

- T_1: Transformation between the *arm_tool_link* frame and the *grasp_frame* created. Obtained with the TF tool of ROS.

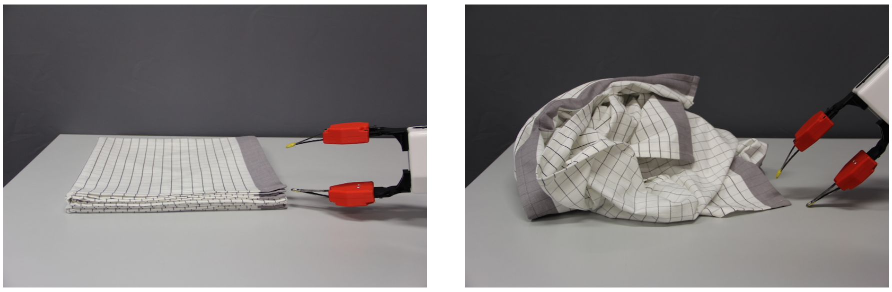
- T_2: Position of the fingertip (*grasp_frame*) with respect to the garment's corner position. It will vary in the three phases, depending on the pre-grasp and post-grasp distance offsets as seen in Fig. 10.

- T_3: Garment's corner position with respect to the robot's mobile platform frame *base_footprint*. Obtained through the perception algorithm.

- T_f: Position of the *arm_tool_link* with respect to the *base_footprint* frame. Is computed with the previous transforms as $T_f = T_3 * T_2 * T_1$.

The T_f transform, represented in red in the figure, is the unknown transform that has to be computed. This transformation gives the relative position of *arm_tool_link* with respect to the robot's base frame in order to bring the fingertip frame to the grasping position. As the grasping is divided into three phases, it is necessary to compute three poses of the *arm_tool_link* in order to move the fingertip to the pre-grasp, grasp and post-grasp positions, which are relative to the garment's corner and are defined by the user (e.g. pre-grasp 10cm away in the X-axis, 0cm away from the Y-axis and 10cm higher in the Z-axis and of the corner).

The difference between both methods lies in the pre-grasp position and in how the end-effector reaches the grasp position from this point. First, when the initial configuration of the garment is folded and the goal is to insert one of the fingertip in between two layers, the best solution is to approach the garment horizontally with the fingertip flat, as seen in Fig. 11a. Second, when the garment is crumpled or lying over the table, the solution can be to perform an environment contact with the modified gripper and the table in order to slide the fingertip below the fabric



(a) Inserting finger between layers. (b) Grasping a corner flat on a table.

Figure 11: Gripper orientation configuration.

in the same way as humans do. To do so, it is best to define the pre-grasp position with the arm more tilted to make the flexible fingertip of the modified gripper first touch and slide over the table sbut preventing the robot arm to collide with it (Fig. 11b).

It is important to remark that with the environment scan previously performed, the occupancy matrix has also saved the garment as an obstacle to avoid. This fact prevents from bringing the robotic arm and its end-effector to the garment's corner as the planning will consider it is in collision. This can be solved by removing the identified object from the environment or defining an allowed collision between the garment and some of the robotic system links. In our case, as the collision will occur with the table in addition to the fabric, it is decided to define a region around the grasping point in which it is allowed collisions with the end-effector. This will include the garment as well as part of the table near the corner, allowing a collision between the fingertip and the table during the grasp position.

4.5 Experiments and Results

As it has been said, in order to obtain accurate measures, an HTC tracker is attached in the garment, enabling on the way a fixed setup which allows repeatibility of the same initial conditions. As this tracker occupies a large area of the garment, making impossible to fix it in the exact position of the garment since it would not allow to grasp it, it is placed at the contrary corner. In addition, due to the flexible properties of the garment, a simple 3D design is used in order to point the exact position of the reference system corresponding to the garment's corner, in order to be able to place the towel in the appropiate configuration. Fig. 12 shows this setup used for performing the experiments, representing the involved reference frames used to obtain the real results.

Hereunder are described the transforms used for obtaining the distance between the garment's corner and the fingertip:

- T_c^{gt}: Distance between the garment's corner and the tracker attached to it.
- T_{gt}^{at}: Transform between the tracker attached to the garment and the tracker mounted over the robotic arm.

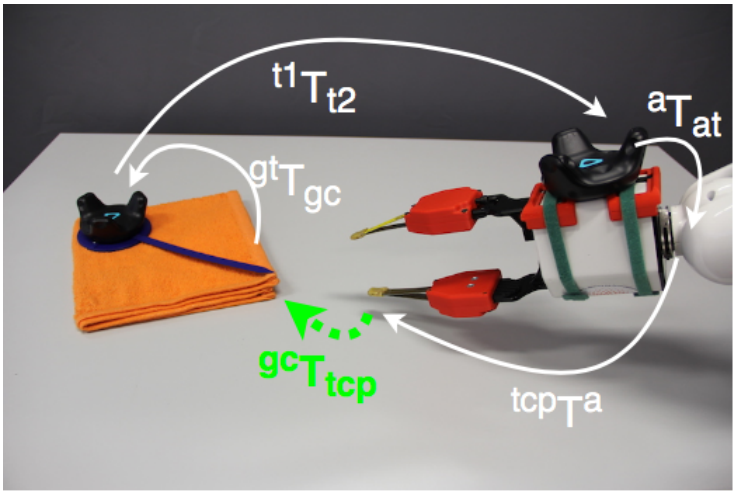


Figure 12: Experimental setup using the HTC Vive system.

- T^{a}_{at}: Relation between the tracker of the arm and the arm itself (*arm_tool_link* frame). This transform is the result of the previous hand-eye calibration.
- T^{tcp}_{a}: TF relation between the *arm_tool_link* frame and the TCP (*grasp_frame* frame.
- T^{c}_{tcp}: This corresponds to the unknown transform that is wanted to obtain. It gives the distance between the fingertip and the garment's corner.

This transform can be computed as the product of the intermediate transforms, in the form:

$$T^{c}_{tcp} = T^{gt}_{c} * T^{at}_{gt} T^{a}_{at} * T^{tcp}_{a} \tag{3}$$

Therefore, to asses the precision of the hand-eye calibration we perform several executions of the application described in order to complete the process of localizing the garment's corner, obtaining its position with respect to the robot and planning the arm motions to bring the fingertip to the corner. Once the robot concludes that it has reached the goal position, ending the motion, the error between the positions of the corner's reference system and TCP frame is obtained as mentioned.

Figures 13 to 15 represent the translation error in millimeters of the end effector positioning at the pre-grasp, grasp and post-grasp positions, measured for 20 trials. These results have been obtained as the difference between the expected position and the final real position. It can be seen that the Y-axis has higher dispersion than the other two axis, meaning a poor positioning in this axis. Nevertheless, the precision is necessary in the Z-axis when it is wanted to insert the gripper's fingertip between two consecutive layers of a folded garment, as in this case.

These results comes from a series of source errors. In first place, TIAGo's hand-eye calibration by itself. As said before, this calibration wants to provide the relation-

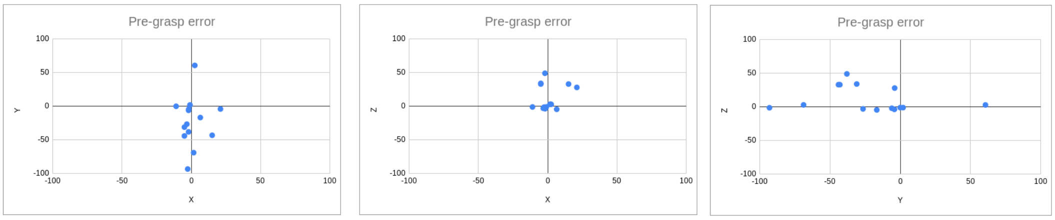


Figure 13: Pre-grasp position errors in millimeters.

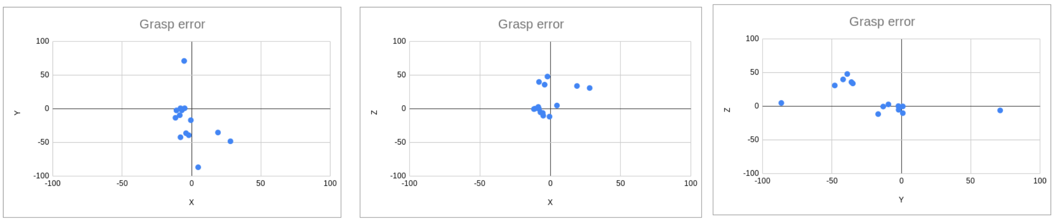


Figure 14: Grasp position errors in millimeters.

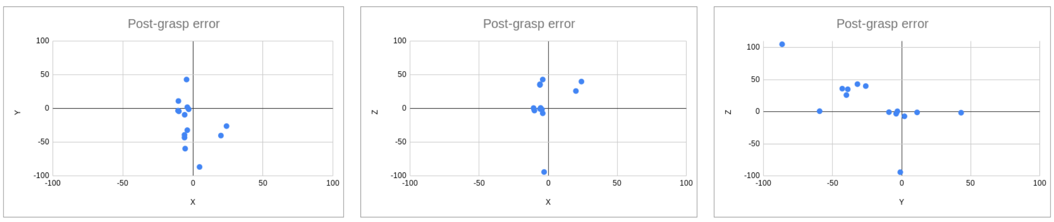


Figure 15: Post-grasp position errors in millimeters.

ship between the robotic arm and the camera, which is obtained through the TF relationship of the robot's reference frame systems. Each of these frames are related to a real part of the robot (e.g. *base_footprint* is related to the center of the mobile platform), which can accumulate error due to the assembling of the pieces. Also, although the HTC system presumes of providing high pose accuracy, it can incorporate noise to the system when the trackers are not correctly sensed by the base stations. This was observed when performing the two additional hand-eye calibrations presented for obtaining the position of the trackers mounted over the robot. In addition, as we use custom made grippers, they do not have a reference system linked in order to obtain its position with respect to other frames of the robot, so we created a reference frame for relating the fingertip of the gripper. As we used a custom made grippers, the difference between the final physical structure of two grippers of the same design can be significant, therefore, the linked reference system must be adjusted during the process. Despite the cumbersome process, this fact can be beneficial as it helps to reduce the error in the precision. Finally, the arm motion planning by itself also has a tolerance in the goal position. Remember that the motion planning used by TIAGo robot computes the planning in order to bring the *arm_tool_link* frame to the position goal and our objective was to precisely move the fingertip to a determinate position. Therefore, despite this tolerance is of small magnitude for the position of the *arm_tool_link*, it is increased for the fingertip position (e.g. a small tolerance in the orientation of the wrist means a higher orientation range in the fingertip pose).

5 Baseline solutions

The present section provides baseline solutions for the two proposed tasks in Section 3, which forms a basis for more complex tasks involving garment manipulation: spreading a tablecloth and folding a towel. This tasks will help at first place to assess the validity of the hand-eye transformations for fine picking of the garment, and in second place to evaluate the performance of dynamic motions for garment manipulation tasks, specially in folding and unfolding garments.

According to [25], most cloth manipulation tasks can be divided into three steps:

1. Location of the object and grasping point selection.
2. Picking the object and performing the preparatory action.
3. Manipulation of the object (e.g. folding or spreading of the fabric).

The first two steps are implemented as explained in Sections 4.3 and 4.4, respectively. The last step, concerning the specific manipulation used to perform a task is presented in Sections 5.1 and 5.2, where the whole process of the proposed tasks (spreading a tablecloth and folding a towel) is explained.

5.1 Spreading a tablecloth

Tasks related to spreading cloth-like objects, such as bed sheets and tablecloths are present in our daily life. For this reason, the first proposed task consists in unfolding a tablecloth and spreading it over a table. We, humans, usually perform this type of tasks by grasping two corners of the fabric and unfolding it in the air due to its large size. Once the first corner is grasped, we perform an edge tracing, that is performe pinching and sliding along the edge until reaching the second corner, then execute a spreading motion in order to extend it in the table.

In order to provide a realistic solution, which could be used with common furnitures in a house, a standard 1.8m long table is selected. To cover it, an IKEA tablecloth of dimensions 1.45 x 2.40 m is used, which is common for this table size. This tablecloth has big dimensions, what implies a challenge in its manipulation for many robots and may need the conception of additional strategies. In addition to the challenge due to the size of the tablecloth, this task presents two other challenges: first, to grasp one single corner of the many layers of the folded garment to allow the following unfolding; and to grasp the second corner performing an unfolding in the air.

Based on human execution of this task, it is conceived as a bi-manual task, which requires two arms to grasp the corners of the tablecloth in a way that allows its manipulation until covering the entire top of a table. For this reason, two mobile manipulator robots TIAGo are used, which mobile platforms gives the advantage to manipulate big pieces of cloth, equipping them with the modified parallel grippers presented in Fig. 2.

Figure 16: Spreading a tablecloth setup.

This task can be divided into three phases that are considered as three different sub-tasks. Each sub-task is dependant of the result of the previous one:

1. Grasping of the first point.
2. Grasping of the second point.
3. Spreading of the tablecloth.

5.1.1 Grasping

This sub-task starts with the cloth placed on top of the table and both robots at the sides of the table as seen in Fig. 16. It is defined from the start of the entire task until the placement of the tablecloth by the first robot in a known configuration and position for the right robot, as relative to Fig. 16, to perform edge tracing. The first robot, located at the left of the table, starts scanning the environment with the head's camera for the detection of obstacles, such as the table. Once it has located the table, it uses the perception algorithm explained in Section 4.3 for locating the garment on top of the table and the grasping position. As it has been said before, this algorithm is slightly modified due to the required setup of the task, since the robot is no longer placed facing the center of the table but on one of its sides, having a corner of the table as the closest point. This leads to select the grasping point of the tablecloth, not as its closest point in the X-axis, but the one with the lowest Euclidean distance. Also, as it will be explained in the following section, the success of the unfolding motion will depend on the tablecloth configuration left at the end of this sub-task. This final configuration of the tablecloth greatly depends on its initial configuration and on the selected grasping point, requiring to grasp the tablecloth not exactly at the corner but in a near edge-point and a further inside of the fabric to ensure a strong grasp, leaving the corner exposed for the edge-tracing motion. This new grasping position is computed by modifying the filtering stage of the perception algorithm accordingly to the described specifications.

Once the pose of the selected grasping point is obtained with respect to the robot's reference frame, the grasping positions of the arm (pre-grasp position, grasp position and post-grasp position) are computed as explained in Section 4.4, enabling then the planning through inverse kinematics which brings in sequence the robotic arm to the positions until grasping the tablecloth with the fingertip. After several experiments of this part of the final system, an issue was detected when grasping the aforementioned tablecloth. Given its big dimensions and the tissue material the tablecloth was too heavy for lifting it with the gripper without loosing the grip since it performed contact only with a small point of the fingertip, being necessary to provide more strength to the gripper. This was solved by adding a rubber band to the top fingertip, which increased the friction and the contact area between the garment and the fingertip but maintained the slippery surface of the bottom fingertip in order to be able to introduce it between the layers of the fabric.

After the tablecloth has been correctly grasped, it is time to bring the tablecloth to a position from which the second robot can perform an unfolding in the air. This position must be between both robots so that both have access to the tablecloth but neither have to reach its arm's extension limit, what could unbalance the robotic platform due to the forces caused by the friction during the sliding motion. At first, TIAGo's arm was sent to this position directly from the post-grasp position but it was seen that the inertia caused by the movement of the arm moved the tablecloth in front of the mobile platform of the second TIAGo. This later prevented to move the robot across the table to spread the tablecloth. To avoid this, the robotic arm should move slower, passing before through an intermediate position. This movement will therefore unfold the folded or crumpled tablecloth by pulling it up and out of the table, letting the excess of fabric fall to the floor in front of the table. The selected offering position is conceived as a predefined joint motion which is reached through a collision avoidance planning taking into account the obstacles present in the environment.

Considering the previous requirements, the success of this sub-task depends on some simplifications; It is assumed that the top layer of the folded garment contains a corner of the tablecloth. In addition, the garment is oriented on the table so that this corner is the closest to the robot. However, the tablecloth can be placed anywhere on the table. Finally, as currently the robot can not distinguish the short edge from the longest one, it is assumed that the tablecloth is folded in such a way to assure that the short edge will be traced.

5.1.2 Unfolding motion: tracing an edge

This part of the task is defined from the end of the previous sub-task until both robots have the tablecloth grasped by the corners and are set to start manipulating the tablecloth for extending it over the table. The main goal of this sub-task is to grasp the second corner of the tablecloth with the second robotic arm. To do so, several approaches are considered; at first, it was considered to perform a fine grasp of the second corner directly from the folded garment in a similar way as done with the first grasping point. This strategy was discarded as it did not suppose an improvement nor difference with the previous sub-task and by that time it was interesting to develop new grasping methodologies and study if we could apply them to other similar tasks. This was not strictly necessary, but we saw it as an opportunity to increase our knowledge in garment manipulation strategies for assistance robotics. For this reason, it was decided to face this problem through the generation of trajectories learned by demonstration to perform an edge tracing of the garment until arriving to the second corner in a similar way as humans do. As far as this author knows, this method has been previously attempted with very small cloths in [12]. In [26], specialized grippers designed to ease this type of manipulation are presented, so before implementing this strategy to the task, it was evaluated if the current design of our gripper allowed to perform this method by attaching garment to a tripod. As our modified gripper was designed with low anti-slippery fingertips, it performed different edge tracing motions straightforward without many issues, so we moved to integrate it to the final system.

The solution consists in grasping a point on the edge of the cloth and from this point start the edge tracing motion, which implies slipping the cloth inside the gripper without loosing it. The second grasping point, marked with a circle in Fig. 17, is assumed to be just under the gripper of the first robot that is already grasping the tablecloth, which as it has been said in the previous section, is placed at a predefined known position. Once the edge of the fabric is grasped, the sliding motion must be executed until reaching the second corner at the end. To do so, we used a Dynamic Movement Primitives (DMP) representation of the motion learned by demonstration. It is decided to perform the edge tracing of the shortest edge of the tablecloth since it implies several simplifications; both robots would be placed closer, what reduces the extension of the robotic arms an therefore the possibility of unbalancing the robots by moving their center of mass; also, a shorter edge implies a shorter trajectory, what simplifies the arm's motion and reduces the risk of loosing the edge. Firstly, a trajectory combining vertical and horizontal motions was implemented. It was seen that the vertical movements helped to recover garment when it was close to being lost and the horizontal movements allowed to move closer to the edge when too much fabric was being grasped. Unfortunately, this trajectory did not work effectively for other type of fabrics (e.g. when the tablecloth is thicker o more rough) as it did not allow to slip easily. New trajectories where evaluated before selecting the most effective one; a vertical straight path executed at constant speed and taking advantage of TIAGo's torso movement to complete the trajectory until the ground. In order to perform this motion, the second TIAGo is placed slightly further away from the table than the first robot, so that it has more mobility in the arm positioning and has access to the tablecloth in a more comfortable way.

Experimentally evaluating the fabric unfolding, it is seen that the sliding trajectory depends on several conditions;

- The initial configuration of the fabric,

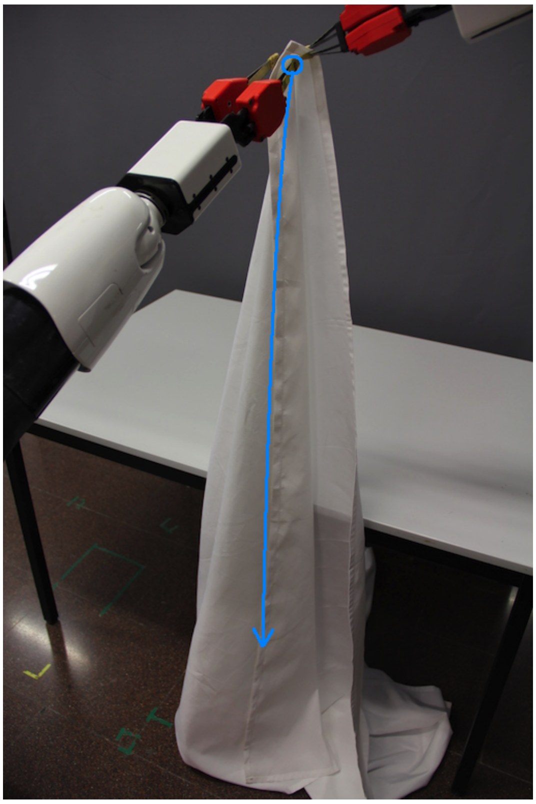


Figure 17: Second grasping point and sliding trajectory.

- the size of the fabric,
- the modified gripper design,
- the material of the fabric,
- the speed of the motion.

Failures on the system are due to this last conditions, which could be solved by, again, adding a tactile sensor to the fingertip for detecting the edge or by using a force sensor to detect if the garment has been lost. In addition, a complex vision algorithm would be useful for the detection of bad initial configurations of the tablecloth after it has been unfolded, such as twistments or tangles that prevents the grasp of the correct edge. The use of this type of sensors will imply an improvement allowing to close the loop of the system by having feedback. This would enable to autonomously adapt the DMP trajectory performance depending on the information provided by the tactile sensor, which could inform for instance if too much fabric is being grasped or if is being close to loose the edge. In addition, the substitution of the gripper for another one, already being in development at IRI [27], that switches between a slippery and an anti-slippery surface could be useful for its use with different tissue materials.

5.1.3 Spreading motion

The last part of this task consists in performing the manipulation of the tablecloth after both corners has been grasped in order to cover the table. This manipulation could be based on human-like motions, teaching the robot trajectories similar to those we execute to extend a tablecloth over a table. It is observed that this type of motions are dependant on the speed at which they are executed as well as on the tissue material properties (e.g. heavy or light, rigid or soft, etc), and are more complex with big pieces of cloth such as the tablecloth used in this project. Thus, it is decided to discard this method and take the advantage that TIAGo's mobile platforms gives. Note that this last strategy is interesting since it is applicable to different sized tables and garments as the bi-manual manipulation is achieved with two independent robots, meaning it can be used in similar tasks such as making a bed. Therefore, after each corner is grasped by a different TIAGo robot both robots move across the table in a straight path to spread the tablecloth on it. The distance moved is dependant to the table's length, which is known. Once the robots have arrived to the end of the table, they perform a movement lowering the arm before opening the gripper in order to release the tablecloth in a way that it does not get sticked in the gripper due to the friction.Grasping

The challenges of this part of the task are several; once the corners are grasped and before moving across the table, it is essential that the remaining tablecloth on the floor is not placed in front of the mobile platforms in order to not go over it. This problem must be solved with the behaviour of the previous sub-tasks, as it has been explained, trying to implement movements that do not lead the tablecloth to this situation. As the robots move at both sides of the table, they must be placed perfectly straight facing the table so that they move in parallel to the long side of the table without deviating, what would tense the tablecloth and loose the grip. Besides, both robots must start the movement at the same time and move at the same constant speed. Also, the initial position of both robots once they have grasped the tablecloth and before moving must avoid the tablecloth from snagging on the table, what would cause strong forces making the grippers loose the garment. This has been solved setting a higher position with the help of the torso, enough for not unbalance the robot during navigation, but sufficient so that the tablecloth passes over the table.

Currently, this sub-task considers some simplifications; the initial position is fixed, so that they can move in a straight line without the need of a navigation planning; also, the table length is known, what defines the amount of displacement of both robots. The coordination between the robots is also simplified, missing the intergration of a synchronization system in order that the robots knows when to start its phase execution autonomously.

The entire process sequence, considering both initial conditions is shown in Fig. 18.

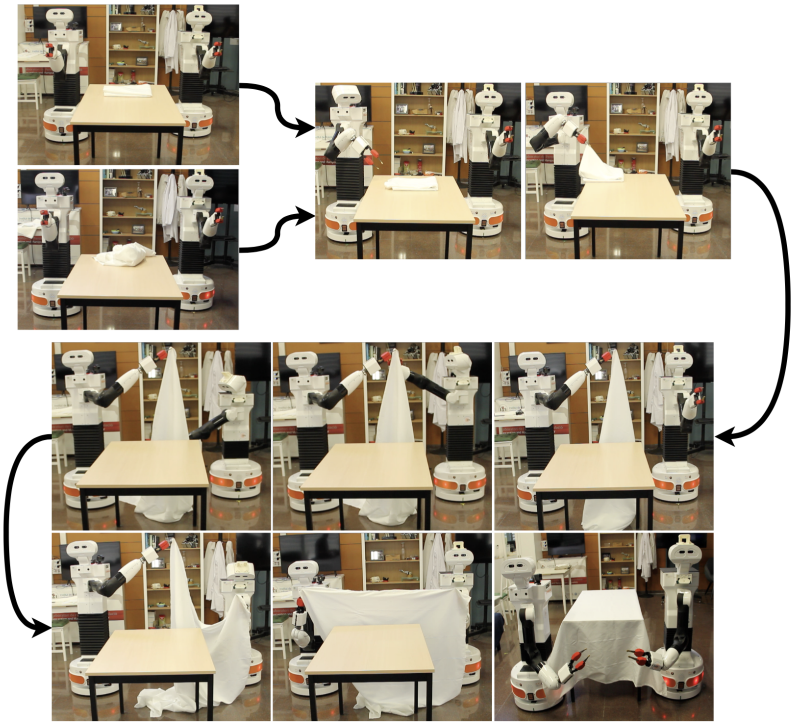


Figure 18: Spreading a tablecloth task execution sequence. First, from the initial configuration (folded or crumpled) one corner is grasped. Then, the other robot follows the edge to locate the second corner and unfolds the garment. Finally, both robots move to spread the tablecloth onto the table.

5.2 Folding a towel

The task of folding a garment is a classic task in cloth manipulation, since it is a repeated action in our daily life for storing and tyding up our cloth items. For this reason, it is commonly found in the robotics community literature, having many different strategies to perform it depending on the available resources and the type and shape of the garment. In works dealing with folding T-shirts or trousers, they usually focus on the visual algorithms used for localizing ideal grasping points (e.g. collars of T-shirts or waistlines of trousers). However, as seen in Section 3.2 there are many works focusing on the folding strategies of napkins, towels or other similar items, which present several forms of folding. For this thesis rectangular towels are used for folding as it is a generic garment and we propose the most common strategy for folding rectangular fabrics: fold by halfs the towel. This can be done by executing a 'folding in the air' method or through the common strategy of placing it flat on a surface such as a table and picking two corners for folding. In this section it is provided a baseline solution based on this last method, which is simpler and more effective for folding small and medium towels, using as in the previous task two TIAGo robots.

The pipeline of this task is dependant on the initial configuration of the towel. We considered two possible starting configurations: crumpled and flat on the table. For the first case, the first steps could be performed as with the tablecloth, that is to say, start grasping the first corner with the first robot and then grasp the second corner with the other robot performing edge tracing, modifying the sliding motion due to the garment's size. The following phase will consist on manipulating the towel for spreading it flat on top of the table, what results in a situation as the second starting configuration. From this position, each robot should grasp a different corner and execute a motion until folding in half the towel, matching two corners of both sides. This last motion can be repeated two or three times, depending on the towel size. Notice that the first phases of this task are similar to the previous task (grasping of the first corner, edge tracing until grasping the second corner and spreading) so for this task these phases will be obviated, focusing on providing solutions for the manipulation phase.

As this task was conceived for the creation of a benchmark in cloth manipulation and as done with the previous task, it is decided to use IKEA standard size towels (0.5 x 1m and 0.3 x 0.4m), which are easy to acquire. The table used in this task is the same as the one used for spreading the tablecloth. In this case, given that the towels are not of big dimensions, the use of the mobile platform for performing the folding is not efficient, as the displacement would be small and would introduce error in the fold. It is considered as a better solution to perform the folds integrally using the robotic arms, placing a robot at each side of the table, as seen in Fig. 19. During the tests of the arm motions for the task, it is observed that with this situation the robot at the right has to be placed farther from the table to have more mobility in the arm in order to reach the towel's corner at his right due that at this position TIAGo robot's arm reaches the limits of its joints.

As with the sliding motion presented in the previous task, the motions for folding the towel will be learned by demonstration. These trajectories should have an arc shape,

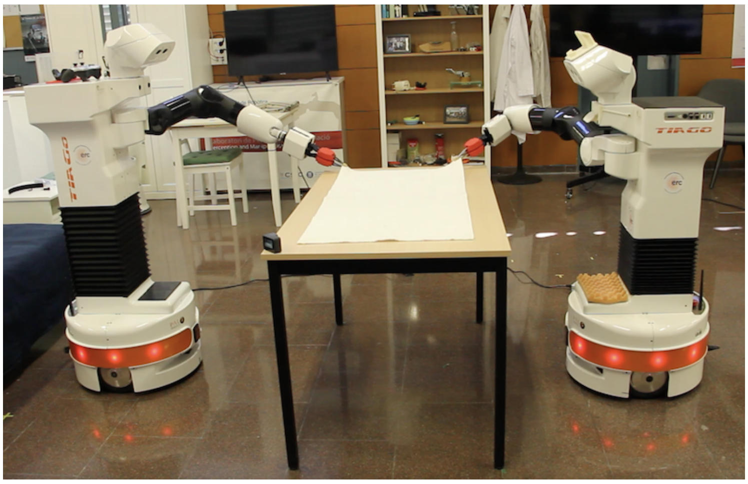


Figure 19: Setup for folding a towel.

separating the towel from the table as the robotic arms move towards the other two corners so that the towel does not slide over the table during the process, fact that happens when moving only in horizontal. As both robots are placed asymmetrically with respect to the table, it can not be used the same motion for both robots as they have to use different joint positions to perform the motion, however the trajectory learned by each robot follows at the same instant of time several points which are at the same height.

As it is a bi-manual task performed by two independent robots, the principal challenges arises in terms of the synchronization between them during the manipulation motion. That is to say, in order to succeed in the result of the folding without undesired bendings or wrinkles, which are caused by small deviations in the movement of the robotic arms trajectories, the folding motion with the two robotic arms must be executed in synchronization, starting and ending at the same moment and passing through the intermediate points at the same instant of time.

The motions learned by the robot assume that the towel's size and the table location with respect to each robot are known. An improvement of the system would consist on adapting these trajectories to different sized towels at different positions depending on the visual feedback.

The sequence of the execution of this task can be seen in Fig. 20.

5.3 Experiments and results

On Section 4 it was studied the precision we can have for performing fine manipulation of cloth-like items. The presented tasks serve as practical demonstrations of the utility of this feature as well as for identifying the missing manipulation skills required for developing more complex applications.

In order to evaluate the performance of the entire system of these tasks, as well as measure the different manipulations involved, some performance metrics including

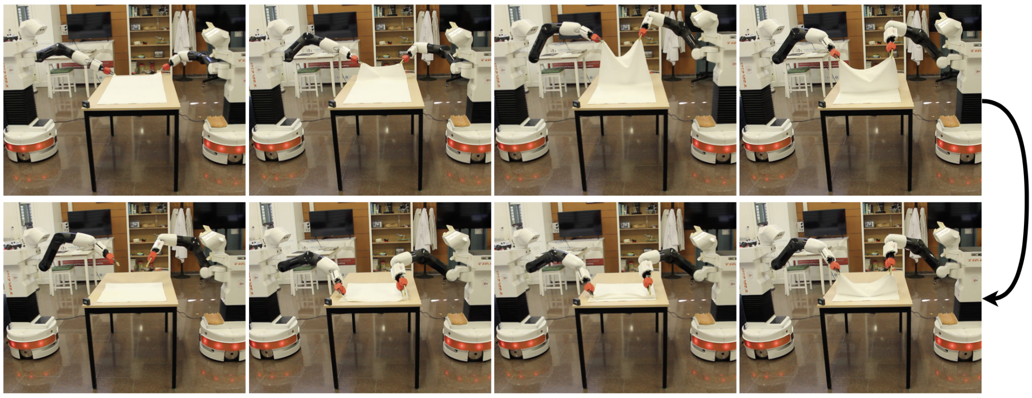


Figure 20: Folding a towel execution process. Only one fold is performed.

success of the different phases of the task, time execution and quality of the cloth resulting configuration are used. Each task is divided into sub-tasks that are evaluated individually, depending on the initial cloth configuration. Therefore, for the spreading task we will have 4 sub-tasks to evaluate, according to the following initial states:

- [**fd**] Cloth is folded over the table.
- [**cr**] Cloth is crumpled over the table.
- [**pg1**] Cloth is pre-grasped by a corner.
- [**pg2**] Cloth is pre-grasped by two points, e.g. is already unfolded.

In addition, each of these sub-tasks can be decomposed into three manipulation phases:

- [**GR1**] Grasping of the first point.
- [**GR2**] Grasping of the second point.
- [**MAN**] Task-specific manipulation of the garment.

In the case of the task of spreading a tablecloth, these phases correspond to the Sections 5.1.1, 5.1.2 and 5.1.3. On the other hand, and as it has been said, in the case of the task of folding a towel we will only evaluate the [**MAN**] phase with initial configuration [**pg2**].

The success in the execution of each of these phases is evaluated in regards of its completeness. That is to say, the success of phases [**GR1**] and [**GR2**] are given in case the grasping points are performed correctly and held during the entire execution. On the other hand, the success of phase [**MAN**] depends on the task: for the spreading task, it is defined as success when the tablecloth fully covers the top of the table; the success of the folding is given when opposing corners of the towel are together after a fold is done. Aside from evaluating the execution time and the completeness of each sub-task, task-specific metrics are defined for evaluating the quality of the result. These specific metrics consist on quality functions that measure the percentage of error of the task result. It measures how much rotated and translated

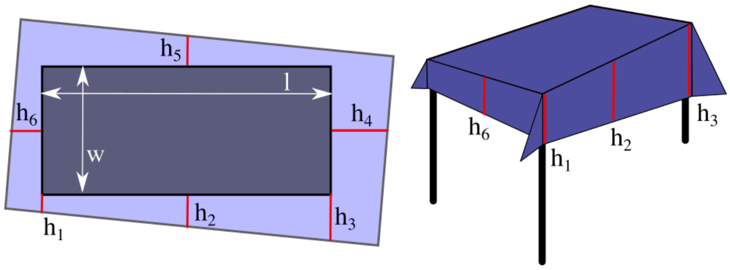


Figure 21: Measures to evaluate the quality of spreading the tablecloth.

is the tablecloth with respect to the table and how perfect is the performed fold. To do so, some measures have to be taken of the resulting cloth configuration.

In the case of the spreading task, we will take 6 tablecloth length at different sides of the table, represented in Fig. 21. These measures are used in the following quality functions, which gives the percentage of error in the tablecloth rotation (Eq. 4), length translation (Eq. 5) and width translation (Eq. 6):

$$E_\alpha = \frac{\arctan(\frac{|h_3 - h_1|}{t_l})}{\pi/4} \tag{4}$$

$$E_l = \frac{|h_6 - h_4|}{c_l - t_l} \tag{5}$$

$$E_w = \frac{|h_2 - h_5|}{c_w - t_w} \tag{6}$$

being t_l and t_w the table length and width and c_l and width c_w the tablecloth length and width.

For the folding towel task, we evaluate the quality of the fold through a ratio between the starting area of the spread cloth and the area after the fold has been performed (see Eq. 7). This is obtained by measuring the length and width of the towel before and after the fold execution, since the area is the product of these two. This quality function gives a 100% error when $A_f = A_i$, that is, no fold has been performed and when $A_f < \frac{A_i}{2}$, meaning that the corner are not matching or that there are wrinkles.

$$E_f = \frac{100}{0.5} \cdot \| \frac{A_f}{A_i} - 0.5 \| \tag{7}$$

Considering all these metrics, we execute 5 trials of each sub-task, acquiring the mentioned measures. We perform a total of 20 experiments (5 trials for 4 sub-tasks) for spreading a tablecloth. The results obtained are collected in Table I.

It can be seen that for the starting configuration **[pg2]** the success rate in the manipulation phase is quite high (80%). The failure on the second trial is caused by

Starting config.	Success [MAN] (1 | 0)	Success [GR2] (1 | 0)	Success [GR1] (1 | 0)	h1	h2	h3	h4	h5	h6	Quality function rotation	Quality function in lenght	Quality function in width	Time in sec
[pg2]	1			42	38.5	35	63	37	55	1.52%	6.67%	2.00%	18.7
	0									0.00%	0.00%	0.00%	
	1			48.5	47	45.5	66.5	29	51	1.33%	12.92%	24.00%	18.1
	1			51	46.5	44	61.5	29	58	1.52%	2.92%	23.33%	18
	1			48	46	44	48	30	60.5	1.41%	10.42%	21.33%	18.3
Summary:	80.00%									1.44%	8.23%	17.67%	avg: 18.28 var: 0.10

Starting config.	Success [MAN] (1 | 0)	Success [GR2] (1 | 0)	Success [GR1] (1 | 0)	h1	h2	h3	h4	h5	h6	Quality function rotation	Quality function in lenght	Quality function in width	Time in sec
[pg1]	1	1		31	17	4	59.5	62	64	1.63%	3.75%	60.00%	72.7
	0	1								0.00%	0.00%	0.00%	72.46
	0	0								0.00%	0.00%	0.00%	
	1	1		22.5	13.5	5.5	57	62.5	62	1.60%	4.17%	65.33%	71.96
	1	1		38	33	24	68	42.5	50.5	1.59%	14.58%	12.67%	71.85
Summary:	60.00%	80.00%								1.61%	7.50%	46.00%	avg: 72.24 var: 0.16

Starting config.	Success [MAN] (1 | 0)	Success [GR2] (1 | 0)	Success [GR1] (1 | 0)	h1	h2	h3	h4	h5	h6	Quality function rotation	Quality function in lenght	Quality function in width	Time in sec
[cr]	0	0	1										
	0	0	0										
	0	0	1										
	0	0	1										
	0	0	1										
Summary:	0.00%	0.00%	80.00%							-	-	-	avg: - var: -

Starting config.	Success [MAN] (1 | 0)	Success [GR2] (1 | 0)	Success [GR1] (1 | 0)	h1	h2	h3	h4	h5	h6	Quality function rotation	Quality function in lenght	Quality function in width	Time in sec
[fd]	0	0	0							0	0	0	
	0	0	0							0	0	0	
	0	0	1							0	0	0	
	0	0	1							0	0	0	
	0	0	1							0	0	0	
Summary:	0.00%	0.00%	60.00%							-	-	-	avg: - var: -

Table I. Performance metrics of spreading tablecloth experiments.

Summary results							
Starting config.	Success [MAN] (1 | 0)	Success [GR2] (1 | 0)	Success [GR1] (1 | 0)	Quality function rotation	Quality function in lenght	Quality function in width	Time in sec
[pg2]	80.00%			1.44%	8.23%	17.67%	avg: 18.28 var: 0.10
[pg1]	60.00%	80.00%		1.61%	7.50%	46.00%	avg: 72.24 var: 0.16
[cr]	0.00%	0.00%	80.00%	-	-	-	avg: - var: -
[fd]	0.00%	0.00%	60.00%	-	-	-	avg: - var: -

Table II. Result summary table of spreading a tablecloth task.

Object	Big towel						
	First fold						
Starting config.	Success [GR1] (1 | 0)	Success [GR2] (1 | 0)	Success [MAN] (1 | 0)	Area before	Area after	Quality function fold	Time in sec
[pg2] - 2 pre-grasps			1	5000	2626	5.04	24.17
			1	5000	2525	1	24.27
			0	5000		-100	24.4
			1	5000	2550	2	24.46
			1	5000	2575.5	3.02	24.44
Summary:			80.00%			2.765	24.348

Table III. Performance metrics of the experiments for folding a towel.

and occasional entanglement of the tablecloth which made the gripper loose the garment. When the starting configuration is **[pg1]** we see that it executes in a 80% of the times the edge-tracing motion correctly and the spreading of the tablecloth has a 60% of success. In this case, one trial failed at the edge tracing as it grasped too much garment, making impossible the manipulation phase. The remaining spreading failure was due to a bad release of the tablecloth. It is observed that the quality in width translation decreases in this sub-task but the other quality errors are very small. Finally, when we start the task having the tablecloth folded or crumpled over the table (**[cr]** and **[fd]**), the entire task can not be executed due to the edge tracing. Nevertheless, the grasping of the first corner (**[GR1]**) has good results. A summary of the results obtained for this task are collected in Table II.

Finally, as it has been said, we execute 5 trials of the manipulation phase of the folding a towel task, that is, starting from the initial configuration **[pg2]**. The measures obtained for these trials are present in Table III.

The results in this task are very good, having an 80% of success with good quality. Only one trial from five failed with an unwanted fold in a corner due to a unforeseen problem with the gripper, preventing that both corners match with its contrary. This bend of the towel was caused in the release step when the corner remained sticked to the fingertip.

Due to the lack of comparison benchmarks in robotic manipulation in general, and in cloth manipulation particularly, we approached the evaluation of these tasks in a systematic way that can be repeatable and measurable. This way, we have defined a benchmark assessing the capabilities of bi-manual systems for cloth manipulation

Assumptions	Used (YES | NO)
Table color	NO
Table position known	YES
Object position known	NO
Tablecloth color	NO
Tablecloth dimensions	YES

Table IV. Assumption used for developing the spreading task.

Figure 23: Different garments used in the development of this thesis.

tasks [28]. A benchmark provides well-defined tasks with standardized characteristics for being reproduced with different robotics systems in order to compare several aspects between them. This benchmark provides metrics to evaluate the strengths and weaknesses of a system and to help comparing different manipulation techniques through the implementation of three cloth manipulation tasks, two of them the described in this thesis. This comparison can be applied also considering the assumptions or simplications that have been used in the development of the task, as for example our assumption, mentioned before are collected in table ??. Benchmarking is a popular topic nowadays in robot manipulation, therefore, the development of the benchmark is an important contribution that has been submitted as an Special Issue about Benchmarking for the journal Robotics and Automation Letters.

5.4 Lessons learned

Due to the practical nature of this section of the project, some lessons can be extracted, which in a future work will help for improving the system.

Experimenting with real robots is hard because involves the interaction of many different modules, from perception to action. Separately, each module can be validated and their limitations and restrictions can be assessed. However, the integration of various modules poses additional challenges, and the evaluation result is not a simple combination of the evaluation of simple modules but something more complex.

Evaluation of robotic tasks is difficult and involves thinking on the concepts of repeatably, the particular robot embodiment. Surprisingly, there are not many benchmarks in robotics to get inspiration from, and no one deals with the problem of textile manipulation. Finding measures that are objective, and that afterwards can be used to compare different approaches is a challenge.

Grasping involves contact, that is collisions, something that usually one would like to avoid in general motion planning of manipulators. We had to make some tricks with the collision map to get the motion planner produce useful paths.

Despite having precision to perform fine graspings, unexpected situations can appear, as for example that the fingertip hits the fabric's border, causing an unwanted bending and preventing the success of the grasp. This means that for performing fine graspings, in addition to the calibration other aspects must be taken into account such as the gripper design, the tissue material properties, etc. This last fact has been continuously observed during the implementation of the project as several types of fabrics (see Fig. 23) have been used to perform the experiments and the same strategy used with one or another garment changed the performance of the execution. Therefore, in addition to the manipulation strategies and techniques designed it is necessary to take into account the garments that are going to be used or the gripper requirements for adapting to different situations. This has inspired the design of a new gripper, able to slip under the clothes due to thin finger endings, and able to can switch between high friction and low friction [27].

The manipulation strategies involve vision to detect the initial configurations of the garments, but the executions of the actions are in open loop as consist in execution of learned motions of different complexities. Some of the skills would benefit of a continuous vision feedback that allows to reshape the action. For example edge tracing would be more effective if the trajectory varies accordingly to the current position of the garment edge.

Other type of sensors would be also beneficial. For example, parts of the task could be improved adding tactile and force sensors in the fingertips for detecting if the garment has been successfully grasped or not and whether it is needed to exert more or less force to hold the fabric. This would allow to enable re-grasps when an issue is found during the grasping, to sense the necessary force in order to grasp or slide the garment between the gripper or to autonomously cancel the execution of the task for instance when the garment is lost during the edge tracing motion.

6 Budget

This section provides the total cost necessary for developing this project, including hardware, software and human resources.

6.1 Hardware ans software costs

As seen in Section 2, as hardware resources we used two TIAGo robots, the HTC Vive system and 3D printed grippers. The price of these resources is presented in the following table:

Resource	**Units**	**Cost (€)**
TIAGo robot	2	50.000
HTC Vive system	1	700
3D impression	1	150

Table V. Prices of hardware resources.

According to Table V, where it shows the price per hour for using these resources, we estimate that the total cost of having at disposal these resorces during the 6 months that this thesis has last is of 27000€.

Resource	**Amortization period (years)**	**Price per hour (€)**
TIAGo robot	5	10
HTC Vive system	1	3.5

Table VI. Hardware resource budget.

On the other hand, the software resources (ROS and MATLAB) are free for students use so the total cost of the softtware used in this project is of 0€.

6.2 Human resources cost

In Table VI the salary of an engineer and of a project manager is shown. It is also presented the total cost in human resources, considering that this project has been developed during 6 months, working approximately 35 hour per week, what results in a total of 840 hours, plus the supervisor work that has dedicated 5 hours a week during the same quantity of months, resulting in 120 hours.

Role	**Cost per hour (€)**	**Hours**	Salary
Project manager	40	120	4.800
Enginyeer	20	840	16.800

Table VII. Human resource costs.

7 Conclusions

The overall objective of this master thesis was to assess the capabilities of the TIAGo robot to perform deformable objects manipulation tasks developing simple manipulation skills. After the bibliographical search we realized that large dimensions clothing had never been approached in previous works; either because large and heavy manipulators are needed or because two robots are needed. We decided to take the challenge and use big towels and large tablecloth, and use two robots.

We have proved that this type of robot can successfully execute autonomous simple tasks involving garment manipulation, as grasping a corner of a garment either introducing accurately a flat and thin custom made gripper between several layers of a folded garment or by pre-colliding with a table for sliding it under the fabric. This application is achieved thanks to a hand-eye calibration which solves the problem of bringing a robotic arm to a determined position that has been previously sensed with a camera. This basic feature allows to sense objects, like textiles, in an uncertain environment and grasp with a robotic arm key points in order to simplify the following pipeline execution. It is clear that the embodiment, and in particular the gripper, plays an important role. The robots in the experiments are equipped with one simple parallel gripper where we have changed the friction of the tips when required. In the future, the evaluation of a new gripper that has been designed in the Perception and Manipulation Lab, taking inspiration from results of this work, will take advantage of the proposed benchmark.

We have also seen that a naive approach as the implemented in this project is sufficient to perform simple sub-tasks separately. Nevertheless, it is not enough to perform in a continuous and effective way complex tasks like the ones presented in this project, which are composed of a combination of different baseline skills. The failures on the final tasks performance suggest a clear issue concerning the open-loop control. We believe that these failures could be solved with the implementation of a closed-loop controller which will help in correcting the behaviour of the robot during the execution of the task. Therefore, we can concluded that in order to improve the performance of the more complex manipulations, as the edge tracing of a garment, it is necessary to incorporate some type of monitoring or sensory feedback to take into account unforeseen situations. This monitoring could be done through visual servoing, or with the incorporation of extra sensors in the grippers such as tactile and force sensors. For example, the use of a tactile sensor during the edge tracing phase of the spreading a tablecloth task would imply a big improvement of the entire task, since the arm motion trajectory could be corrected depending on the information provided by the sensor.

When designing the experiments, we realized that it was very difficult to find a comparison basis with other methods. This is in part because benchmarking in robotics in general is hard due to the complexity of performing experiments, and in defining clear initial conditions that can be replicated independently of the system used. But also, because textile manipulation in particular is a hard problem. Thus, we made an effort to propose a clear set of representative tasks and initial configurations that can be replicated and compared among different robotic systems. This in turn has been the base for proposing a benchmark for evaluating manipulation of textiles that has been submitted for publication in a robotics journal [28]. Our benchmark still has some limitations that should be overcomed in the future. The main one is that a better way of specifying the initial position of the garments has to be developed. This is complex, as crumpled garments are very difficult to specify and the exact initial position will not be attained.

References

[1] P. Rashidi and A. Mihailidis, "A survey on ambient-assisted living tools for older adults," *IEEE journal of biomedical and health informatics*, vol. 17, no. 3, pp. 579–590, 2012.

[2] T. L. Mitzner, T. L. Chen, C. C. Kemp, and W. A. Rogers, "Identifying the potential for robotics to assist older adults in different living environments," *International journal of social robotics*, vol. 6, no. 2, pp. 213–227, 2014.

[3] J. Borràs, G. Alenya, and C. Torras, "A grasping-centered analysis for cloth manipulation," *(submitted) arXiv preprint arXiv:1906.08202*, 2019.

[4] K. Hamajima and M. Kakikura, "Planning strategy for task of unfolding clothes," *Robotics and Autonomous Systems*, vol. 32, no. 2-3, pp. 145–152, 2000.

[5] P. Monsó, G. Alenyà, and C. Torras, "Pomdp approach to robotized clothes separation," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2012.

[6] A. Ramisa, G. Alenya, F. Moreno-Noguer, and C. Torras, "Using depth and appearance features for informed robot grasping of highly wrinkled clothes," in *IEEE Int. Conf. on Robotics and Automation*, pp. 1703–1708, 2012.

[7] A. Doumanoglou, J. Stria, G. Peleka, I. Mariolis, V. Petrik, A. Kargakos, L. Wagner, V. Hlavac, T.-K. Kim, and S. Malassiotis, "Folding clothes autonomously: A complete pipeline," *IEEE Transactions on Robotics*, vol. 32, no. 6, pp. 1461–1478, 2016.

[8] J. Maitin-Shepard, M. Cusumano-Towner, J. Lei, and P. Abbeel, "Cloth grasp point detection based on multiple-view geometric cues with application to robotic towel folding," in *IEEE Int. Conf. on Robotics and Automation*, pp. 2308–2315, 2010.

[9] C. Bersch, B. Pitzer, and S. Kammel, "Bimanual robotic cloth manipulation for laundry folding," *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1413–1419, 2011.

[10] Y. Li, D. Xu, Y. Yue, Y. Wang, S.-F. Chang, E. Grinspun, and P. K. Allen, "Regrasping and unfolding of garments using predictive thin shell modeling," in *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pp. 1382–1388, IEEE, 2015.

[11] M. Cusumano-Towner, A. Singh, S. Miller, J. F. O'Brien, and P. Abbeel, "Bringing clothing into desired configurations with limited perception," in *IEEE Int. Conf. on Robotics and Automation*, pp. 3893–3900, 2011.

[12] H. Yuba, S. Arnold, and K. Yamazaki, "Unfolding of a rectangular cloth from unarranged starting shapes by a dual-armed robot with a mechanism for managing recognition error and uncertainty," *Adv. Robot.*, vol. 31, no. 10, pp. 544–556,

2017.

[13] B. Balaguer and S. Carpin, "Combining imitation and reinforcement learning to fold deformable planar objects," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 1405–1412, 2011.

[14] L. Sun, G. Aragon-Camarasa, S. Rogers, and J. P. Siebert, "Accurate garment surface analysis using an active stereo robot head with application to dual-arm flattening," in *IEEE Int. Conf. on Robotics and Automation*, pp. 185–192, 2015.

[15] S. Cuén Rochín, J. Andrade-Cetto, and C. Torras, "Action selection for robotic manipulation of deformable objects," in *Frontier Science Conference Series for Young Researchers: Experimental Cognitive Robotics (FSCYR: ECR)*, ESF-JSPS, 2008.

[16] B. Willimon, S. Birchfield, and I. Walker, "Model for unfolding laundry using interactive perception," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 4871–4876, 2011.

[17] Y. Li, Y. Yue, D. Xu, E. Grinspun, and P. K. Allen, "Folding deformable objects using predictive simulation and trajectory optimization," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 6000–6006, 2015.

[18] S. Miller, J. Van Den Berg, M. Fritz, T. Darrell, K. Goldberg, and P. Abbeel, "A geometric approach to robotic laundry folding," *The International Journal of Robotics Research*, vol. 31, no. 2, pp. 249–267, 2012.

[19] A. X. Lee, H. Lu, A. Gupta, S. Levine, and P. Abbeel, "Learning force-based manipulation of deformable objects from multiple demonstrations," in *IEEE Int. Conf. on Robotics and Automation*, pp. 177–184, 2015.

[20] S. Miller, J. van den Berg, M. Fritz, T. Darrell, K. Goldberg, and P. Abbeel, "A geometric approach to robotic laundry folding," *The International Journal of Robotics Research*, vol. 31, no. 2, pp. 249–267, 2012.

[21] Y. Li, Y. Yue, D. Xu, E. Grinspun, and P. K. Allen, "Folding deformable objects using predictive simulation and trajectory optimization," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 6000–6006, 2015.

[22] A. Colomé and C. Torras, "Dimensionality reduction for dynamic movement primitives and application to bimanual manipulation of clothes," *IEEE Transactions on Robotics*, vol. 34, no. 3, pp. 602–615, 2018.

[23] R. Y. Tsai and R. K. Lenz, "A new technique for fully autonomous and efficient 3d robotics hand/eye calibration," *IEEE Transactions on robotics and automation*, vol. 5, no. 3, pp. 345–358, 1989.

[24] R. Horaud and F. Dornaika, "Hand-eye calibration," *Int. J. Robot. Res.*, vol. 14, no. 3, pp. 195–210, 1995.

[25] J. Sanchez, J.-A. Corrales, B.-C. Bouzgarrou, and Y. Mezouar, "Robotic ma-

nipulation and sensing of deformable objects in domestic and industrial applications: a survey," *The International Journal of Robotics Research*, vol. 37, no. 7, pp. 688–716, 2018.

[26] K. S. M. Sahari, H. Seki, Y. Kamiya, and M. Hikizu, "Edge tracing manipulation of clothes based on different gripper types," *Journal of Computer Science*, vol. 6, no. 8, pp. 872–879, 2010.

[27] S. Donaire, J. Borràs, G. Alenyà, and C. Torras, "A versatile gripper for cloth manipulation," 2020.

[28] I. Garcia-Camacho, M. Lippi, M. C. Welle, H. Yin, R. Antanova, A. Varava, J. Borràs, C. Torras, A. Marino, G. Alenyà, and D. Kragic, "A benchmark for dual-arm cloth manipulation," *(submitted)*, 2019.