Bachelor’s thesis

Czech
Technical
University
in Prague

F3

Faculty of Electrical Engineering
Department of Cybernetics

Extraction of Features from Moving Garment

Michal Neoral
CYBERNETICS AND ROBOTICS, Robotics

May 2014
Supervisor: Ing. Pavel Krsek, Ph.D.

Czech Technical University in Prague
Faculty of Electrical Engineering

Department of Cybernetics

BACHELOR PROJECT ASSIGNMENT

Student: Michal Neoral

Study programme: Cybernetics and Robotics

Specialisation: Robotics

Title of Bachelor Project: Extraction of Features from Moving Garment

Guidelines:

1. Study the process of creating dynamic physical models of the fabric.
2. Study the methods of measuring and realization of experiments on the CloPeMa robotic workplace.
3. Design a method of measurement and extraction of image features. The features will be used to estimate the parameters of the dynamic model of garment.
4. Implement the proposed method in Matlab and necessary part of the code in the ROS environment.
5. Prepare a data set that will be used to validate the methods for estimation of the model parameters.
6. Prepare adequate documentation.

Bibliography/Sources:

[1] Šonka, Milan; Hlaváč, Václav, and Boyle, D. Roger: Image Processing, Analysis and Machine Vision. Thomson, Toronto, Canada, 3 edition, April 2007. ISBN 978-0-495-08252-1.

[2] Bender, Jan; Deul, Crispin: Adaptive cloth simulation using corotational finite elements. Computers & Graphics-UK 37 (7),: 820-829, DOI 10.1016/j.cag.2013.04.008, 2013.

[3] Lee, Yongjoon; Ma, Jaehwan; Choi, Sunghee: Automatic pose-independent 3D garment fitting. Computers & Graphics-UK, 37 (7), 911-922, DOI: 10.1016/j.cag.2013.07.005, 2013.

[4] Kang; Liu, Yue; Ogunmakin, Gbolabo; et al.:Panoramic Gaussian Mixture Model and large-scale range background substraction method for PTZ camera-based surveillance systems. Machine Vision and Applications, 24 (3), 477-492,DOI: 10.1007/s00138-012-0426-4, 2013.

Bachelor Project Supervisor: Ing. Pavel Krsek, Ph.D.

Valid until: the end of the summer semester of academic year 2014/2015

L.S.

doc. Dr. Ing. Jan Kybic
Head of Department

prof. Ing. Pavel Ripka, CSc.
Dean

Prague, January 10, 2014

České vysoké učení technické v Praze
Fakulta elektrotechnická

Katedra kybernetiky

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Student: Michal Neoral

Studijní program: Kybernetika a robotika (bakalářský)

Obor: Robotika

Název tématu: Získání příznaků z obrazu pohybující se látky

Pokyny pro vypracování:

1. Seznamte se s postupem vytváření dynamických fyzikálních modelů látky.
2. Seznamte se s možnostmi měření a realizace experimentů na robotickém pracovišti projektu CloPeMa.
3. Navrhněte metodu měření a získání obrazových příznaků, které lze použít pro odhad parametrů dynamického fyzikálního modelu látky.
4. Implementujte navrženou metodu v prostředí Matlab a potřebné části v prostředí ROS.
5. Připravte datovou sadu, která bude použita při ověřování metod odhadování parametrů fyzikálního modelu.
6. Vše pečlivě zdokumentujte.

Seznam odborné literatury:

[1] Šonka, Milan; Hlaváč, Václav, and Boyle, D. Roger: Image Processing, Analysis and Machine Vision. Thomson, Toronto, Canada, 3 edition, April 2007. ISBN 978-0-495-08252-1.
[2] Bender, Jan; Deul, Crispin: Adaptive cloth simulation using corotational finite elements. Computers & Graphics-UK 37 (7),: 820-829, DOI 10.1016/j.cag.2013.04.008, 2013.
[3] Lee, Yongjoon; Ma, Jaehwan; Choi, Sunghee: Automatic pose-independent 3D garment fitting. Computers & Graphics-UK, 37 (7), 911-922, DOI: 10.1016/j.cag.2013.07.005, 2013.
[4] Kang; Liu, Yue; Ogunmakin, Gbolabo; et al.:Panoramic Gaussian Mixture Model and large-scale range background substraction method for PTZ camera-based surveillance systems. Machine Vision and Applications, 24 (3), 477-492,DOI: 10.1007/s00138-012-0426-4, 2013.

Vedoucí bakalářské práce: Ing. Pavel Krsek, Ph.D.

Platnost zadání: do konce letního semestru 2014/2015

L.S.

doc. Dr. Ing. Jan Kybic
vedoucí katedry

prof. Ing. Pavel Ripka, CSc.
děkan

V Praze dne 10. 1. 2014

Acknowledgement / Declaration

Foremost, I would like to thank Ing. Pavel Krsek, Ph.D. for his great guidance and help throughout this project. I would also like to thank Ing. Libor Wagner and Bc. Vladimír Petrík for their help with the robot experiments. Finally, I would like to thank my parents for their support during my studies.

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

V Praze dne 5. 5. 2013

....

Abstrakt / Abstract

Cílem této práce je navržení metody pro získání obrazových příznaků z pohybující se látky. Tyto příznaky by měly být dále využity pro získání parametrů dynamického modelu látky.

Pro účely získání parametrů dynamického modelu sledujeme dynamické chování reálné látky. Toto dynamické chování je vyvoláno krátkým a jednoduchým pohybem robotického manipulátoru, který drží tuto látku v chapadle.

Zaměřili jsem se převážně na získání těchto parametrů z časové posloupnosti dat z hloubkového senzoru a standardní barevné kamery, jimiž je pohyb látky snímán.

Vytvořené řešení by mohlo v kombinaci s modelem látky poskytnout lepší možnosti simulace pohybu, plánování pohybu a detekci kolizí při robotické manipulaci s látkami.

V rámci této práce byla pořízena obrazová data sestávající se z čtyřiceti šesti měření pro osm různých látek. Metoda výpočtů byla ověřována na těchto datech.

Provedené experimenty ověřily, že navržená metoda má dostatečnou opakovatelnost.

Klíčová slova: dynamický model; model oděvu, textilie; extrakce příznaků; 3D obraz; silueta.

Překlad titulu: Získání příznaků z obrazu pohybující se látky

The point of this work is to design a method to obtain image features of the moving garment. This features should be further used to estimate the parameters of the dynamic model of the garment.

For the purpose of obtaining the parameters of the dynamic model, we observed the dynamic behavior of the real garment. This dynamic behavior is caused by a short and simple motion of a robotic manipulator which holds the garment in the gripper.

We concentrated mainly on the obtaining these parameters from time series of data from the rangefinder sensor and from a standard color camera, by which is the motion of the garment captured.

Designed solution could, combined with a model of the garment, provide better possibilities for motion simulation, motion planning and collision detection in robotic manipulation with the garments.

As part of this work, I captured image data consisting of forty-six measurements for eight different types of the garment. The method of computation was verified on that data set.

The experiments verified that the proposed method has satisfactory repeatability.

Keywords: dynamic model; garment model; feature extraction; 3D image; silhouette.

Contents /

Tables / Figures

Chapter 1
Introduction

1.1 Motivations

This bachelor thesis is a part of Clothes Perception and Manipulation project (CloPeMa, 2012-2015) funded by the European Commision. CloPeMa is research project which aims to advance the state of the art in the autonomous perception and manipulation of fabrics, textiles and garments. The CLoPeMa robot will learn to manipulate, perceive and fold a variety of textiles [1].

The CloPeMa project is based on the manipulation of clothes (garments). Simplified dynamic physical model of real garment is useful for the manipulations e.g. for simulating motion, motion planning or collision detection. There are lots of models that could be used to create a virtual model of the garment[1]). The creating virtual model depend on the garments parameters. We need to obtain model of a real garment, so we need to create it based on the real parameters. To find parameters of a real garment, some methods are used, which are mainly based on a mechanical stress studied textile or clothing.

1.2 The State of the Art

The main sphere of using dynamic simulation of garment is computer graphic. These simulations are mainly for a realistic look, but not for real dynamic physical behavior of the garment [2] (including modern metod of simulating like [3–4]). A lot of methods have been developed in the field of simulation models of garments (especially fabrics) [5–11]. Some of these tools or methods of simulation use for construct model a parameters of real garments (fabrics) [9, 11].

In the science and industry exist several measuring techniques which is used to find elementary parameters of fabrics e.g. KESF, FAST or FAMOUS. Kawabata's Evaluation System of Fabric (KESF) is used to get the mechanical properties of the clothes. KESF contains a several equipments for measure these properties. KESF was developed for mass-spring method (from [12]). The method need a piece of fabric (size depends on the current implementation) for the measurement. On this sample is applied a force in the different directions. The KESF produces graphs of change dimensions depending on the applied force. The KESF instruments test with high accuracy following parameters: compression, pure bending rigidity, roughness, shear, surface friction and tensile [12–14].

Very similiar to the Kawabata's System (KESF) is the most popular commercial systeme - Fabric Assurance by Simple Testing (FAST). Both systems were designed to measure fabric mechanical properties at low-stress level, but both systems use different testing principles. KESF system measure deformation depending on the increasing and decreasing applied force while FAST system determines deformation level

[1]) Create a dynamic physical model of the garment is not the point of this work.

at a single point on the deformation curve, therefore FAST system cannot measure hysteresis [14]. Another differences are that the KESF use different equipment for each property. The FAST measure more properties on one equipment, so the number of equipments are reduced [13].

The other method Fabric Automatic Measurement and Optimisation Universal System (FAMOUS) is faster method of measurement based on mechanical stressing of the garment than KESF or FAST. A complete suite of measurement take less than five minutes [13].

All these measuring techniques were designed for using in textile and clothes industry, but also are used for computer graphics simulation of garments. These techniques measure e.g. flexural rigidity, shear, surface, compression or tensile properties, but need tens of measurements equipments and process to acquire parameters. Process takes from a few minutes (FAMOUS) up to units of hours (KESF) [12, 15–16, 13]. While existing methods give excellent results and detailed description of the garment fabric, but do not tell us anything about the whole garment. Moreover, these methods are slow and expensive.

There are also methods of estimating cloth simulation parameters based on extraction features from video (e.g. [17]). This method is based on the fabric projected a structured light pattern of horizontal stripes. A perceptually motivated metric based on matching between folds is used to compare video of real cloth with simulation. This metric compares two video sequences of cloth and returns a number that measures the differences in their folds [17].

The work [17] follows an experimental work using a MOCAP system with twelve cameras [18]. The method uses the mass-spring system is described in the [8]. The Kawabata Evaluation System (KESF) is used to obtain mass and springs parameters and the damping parameters are computed from free fall motion of the garment. This method is the most precise method of the aforementioned. However, this method uses KESF and MOCAP, thus is expensive and slow. Moreover, this method is primary used for animation realism [18].

1.3 Goals

Our goal is to get the parameters of the model behavior of the garment. Detailed examination of the garment is technically and time-consuming, which is unsuitable for estimating the model of the garment in handling, in our case.

Therefore, we propose which parameters we will need for build a simple dynamic physical model and we propose easiest way to obtain these parameters. To do that, we need the real garment track and simply describe it so that it can be compared with the model of the garment. At the same time we want from this comparison to determine the parameters of the garment. The point of this thesis is to tracking the behavior of real garment. The behavior (i.e. the motion) of the garment will be in the limited case under controlled conditions using for a description of features. This motion causes the robot and we will capture the motion according to available equipment of robot (chap. 2.1), thus we use the RGB camera and rangefinder.

Chapter 2
Description of Manipulator

2.1 Workplace

2.1.1 Manipulator

The testbed is composed of two robotic manipulator arm Motoman MA1400 (showed in the figure 2.1). First arm is called as `r1` (or also appears as `R1`). Second arm is similarly marked `r2` (`R2`). The arms `r1` and `r2` are placed on the turntable. The turntable is marked as `external axis` (or `Ext.` or possibly as axis `13`). Location of arms and `Ext.` axis of turntable can be better seen from (figure 2.2).

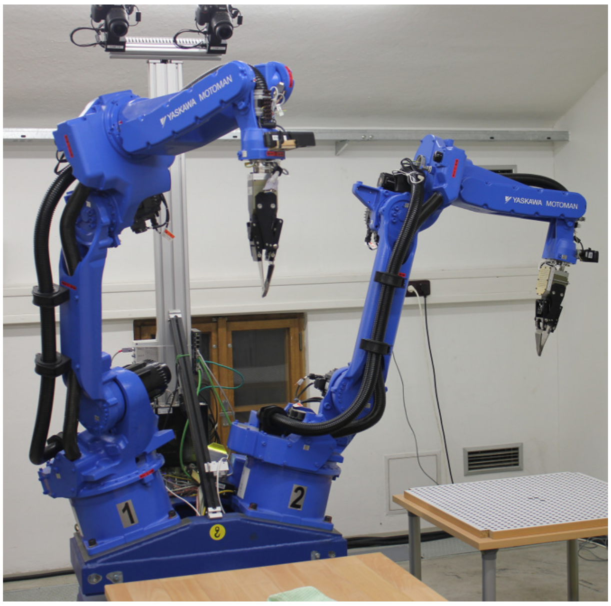


Figure 2.1. Manipulator of CloPeMa project location at CTU

Each arm of manipulator has 6 rotation axes. The axes are labeled according to the manufacturer with the letters `S`, `L`, `U`, `R`, `T` and `B` (figure 2.3). This is the description of single arm of robot. Numeral is added to identify the arms e.g. `S` axis located on the arm `r1` will be called `S1`, etc. Similarly to the designation of arms we can meet even using small letters (eg.: `s1`).

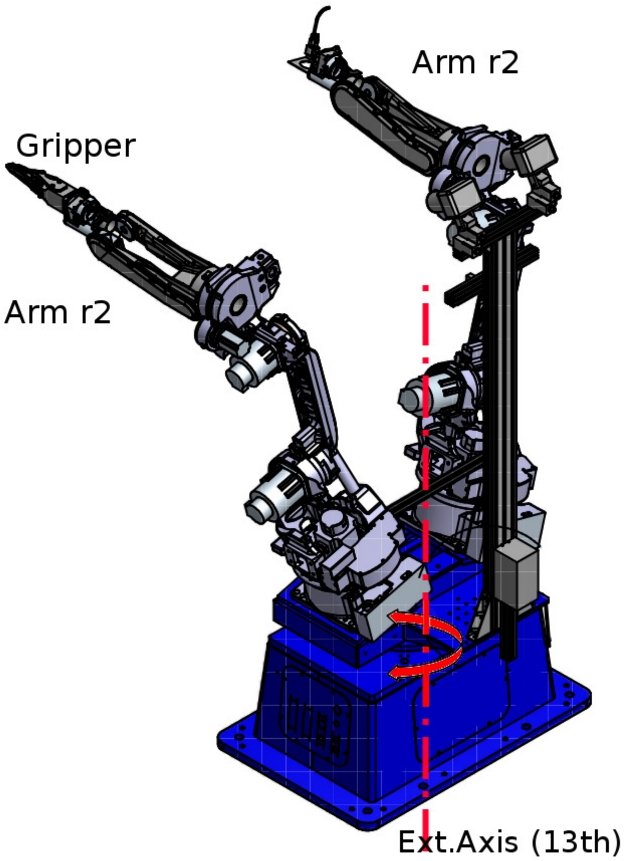


Figure 2.2. Identification of arms and location of external axis (from [19]).

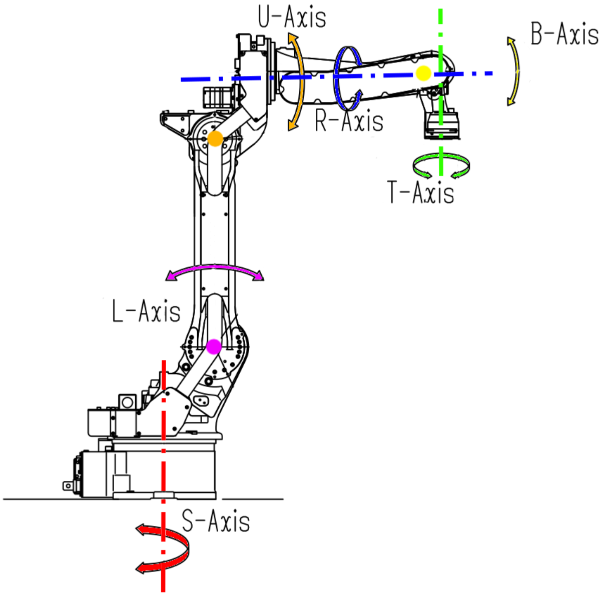


Figure 2.3. Description of a robotic arm Motoman MA1400 - axis (from [20]).

2.1.2 End Effector

Each of arms r1 and r2 are ended with eletricly controlled grippers (figure 2.4). Grippers are designed for grasping garment. The prototype of gripper was developed especialy for CloPeMa project by the colleagues from University of Genoa. The gripper is composed of two flexible fingers. Flexible fingers are used for lifting garments from the table. Gripper has a tactile sensor at the ”finger tips” to sense the garment material using little rubbing motions between the ”gripper finger” [1].

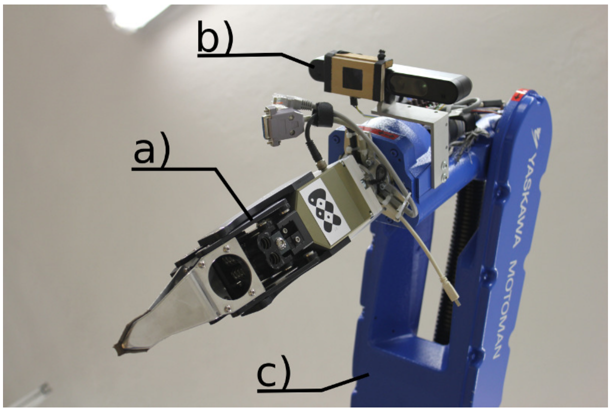


Figure 2.4. End effector (gripper). a) gripper, b) sensor Asus Xtion, c) end of arm on which the gripper is mounted.

2.1.3 Sensors

The robot has been equipped with a variety of sensors:

- Robot Binocular-Vision Head
  - The robot head comprises two Nikon D5100 DSLR cameras. The cameras are mounted on pan and tilt units. The head provides the robot system with high resolution 3D points clouds [1].
- Photometric Stereo Gripper-Mounted Sensor
  - The photometric stereo sensor is a small scale close range camera sensor. The sensors captures at 1280×800 px resolution and has software support to do 3D reconstruction of close-range garment surface [1].
- Wrist Force-Torque Sensor
  - ATI Mini45 FT six-axis force/torque sensor is integrated in the wrist of one gripper. It is used to sense for contact of the gripper with table and for feedback for the robot system when stretching out a held-up piece of clothing [1].
- Xtions
  - Asus Xtion Pro Live sensor is able to record RGB images and depth maps. The resolution of RGB images is up to 1280×1024 px and resolution of the depth images is up to 640×480 px with 30 fps [21]. The robot has three these sensors — one on the each arm [1] and one on the turn table. Sensor location on the manipulator arm shown in figure 2.4.

Xtion sensor is for our purposes the most suitable. Xtion has sufficient rosolution of RGB image and depth map. In addition, Xtion has a sufficient number of frames per second (compared to slower stereo-pair). Xtion mounted on the arm `r1` is called `xtion1` and Xtion mounted on the arm `r2` is called `xtion2`. Position of cameras is shown in figure 2.4.

2.2 Software

Robot is operated using Robot Operating System (ROS). ROS is an open-source system. ROS is not an operating system in the traditional sense of process management and scheduling. Rather, it provides a structured communications layer above the host operating systems of a heterogenous compute cluster [22]. In CloPeMa project is used Ubuntu (Debian-based Linux OS) as a host operating system.

2.2.1 Robot Operating System

A system built using ROS consists of a number of processes, potentially on a number of different hosts, connected at runtime in a P2P topology. The fundamental concepts of the ROS implementation are **nodes**, **messages**, **topics**, and **services**.

Nodes are processes that perform computation. ROS is designed to be modular. A system is typically comprised of many nodes. In this context, the term ”node” is interchangable with ”software module”. Nodes communicate with each other by passing messages. A **message** is a a strictly typed data structure. Standard primitive types (integer, floating point, boolean, etc.) are supported. Arrays of primitive types and constants are supported too. Messages can be composed of other messages, and arrays of other messages, nested arbitrarily deep. A node sends a message by publishing it to a given **topic**. A node that is interested in a certain kind of data will subscribe to the appropriate topic. There may be multiple concurrent publishers and subscribers for a single topic, and a single node may publish and/or subscribe to multiple topics. In general, publishers and subscribers are not aware of each others existence [22].

Although the topic-based publish-subscribe model is a flexible communications paradigm, its ”broadcast” routing scheme is not appropriate for synchronous transactions, which can simplify the design of some nodes. In ROS, we call this a **service**, defined by a string name and a pair of strictly typed messages: one for the request and one for the response [22].

In the ROS are designed a large number of tools e.g. for get and set configuration parameters, for plotting or visualisation. For this project is important a **rosbag tool**. This is basically a set of tools for recording from and playing back to ROS topics [23]. Using this tool we can record all or just some chosen topics. The chosen topics are passed as parameters to the rosbag tool. The rosbag tool records all published messages from chosen topics, including timestamp. The tool stores timestamped messages to a specially formatted `*.bag` file (also `bagfile` or `rosbag-file`). The rosbag tool can later replay these messages from rosbag file.

Chapter 3
Capturing data

3.1 Goals of measurement

The requirement on the measurement is to obtain mathematical features by which could be used to estimate the parameters of the dynamic physical model of garment. These features we determine by tracking hanging garment. Motion of hanging garment will be caused the motion of the manipulator gripper that holds garment. We have chosen the simplest motion, which could give us the necessary data. Data that could be used to obtain the features of the dynamic model of garment. We use the motion of garment in a plane, ideally excited by moving gripper of a garment in a straight line (line segment) in the plane. Based on the sensors that we have available, we have choose types of motion capturing. We decided capture the motion of garment in two ways. We use both possibilities of Xtion sensor:

a) Standart RGB video camera captures a silhouette of the garment against the constant background when garment is moving **perpendicular to the optical axis**.
b) Standart rangefinder captures the garment when garment is moving **along the optical axis**.

3.2 Realisation

Both arms of manipulator are used for the experiment. One arm causes a motion of garment that is held by the gripper of this arm. Xtion sensor, located on the second arm, is used for capturing the motion.

Motion of the gripper along a straight line or line segment (chap. 3.1) requires synchronous motion of the all manipulator axes. This motion is ensured that so the end of gripper moves on the selected points on the line segment. If robot is controlled by ROS, we found that the dynamics of the manipulator is not fast enough to perform the desired motion of the gripper with garment necessary speed.

However, it is possible to achieve the required speed when the motion is based on a motion of a one axis. The synchronous motion of all axes was replaced by a motion a one axis, which may not be synchronous with the other. Thus the motion of gripper along line segment is approximated by moving the gripper on the part of the circle.

Another limitation is a workspace, such that it is not possible to place the camera `xtion` in the appropriate position to capture RGB images (ie, the position where the gripper with garment moves perpendicular to the optical axis) and then the camera `xtion` move to position suitable for capturing depth maps (ie, the position where the gripper with garment moves along the optical axis). These restrictions are solved via camera `xtion` position (ie the position of the arm with the camera) which is fixed in the same position for record RGB videos as well as for sensing depth maps. Instead, the arm with garment makes a move of gripper with two different ways so that the motions fulfilled the conditions for sensing with each sensors (perpendicular position vs. along the optical axis whith is shown it the figure 3.1).

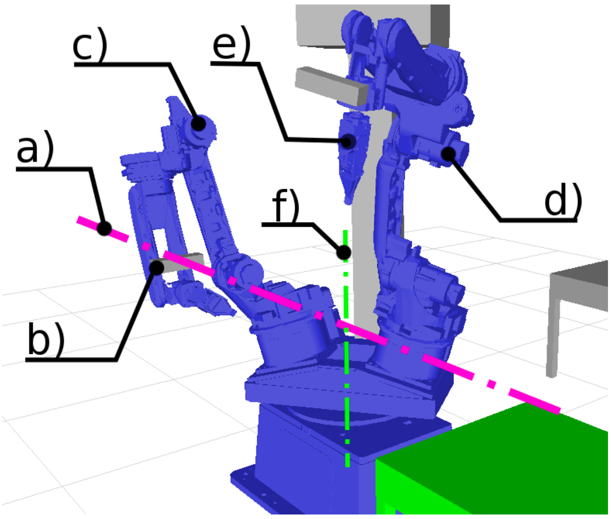


Figure 3.1. Position of arm with camera. a) optical axis of camera xtion1, b) camera xtion1, c) arm r1, d) arm r2, e) gripper of arm r2, f) ext. axis.

3.3 Capturing of RGB

3.3.1 External Axis

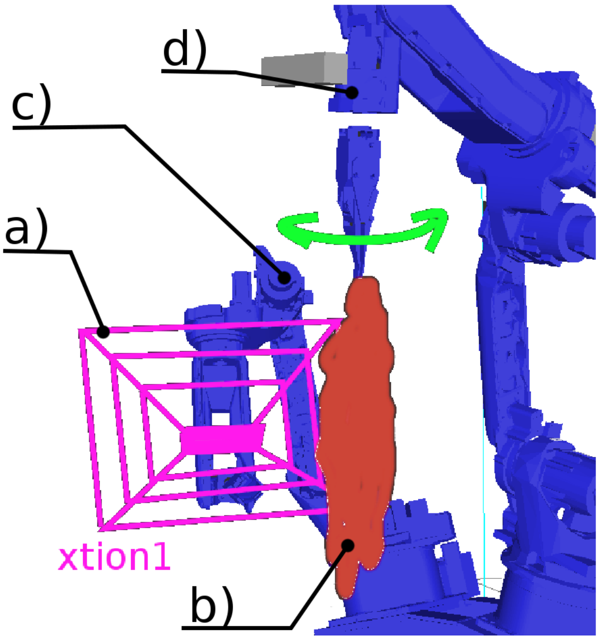


Figure 3.2. Suggestion of motions of gripper with garment perpendicular to optical axis. a) mooted of field of vision of camera xtion1, b) garment, c) arm r1, d) arm r2.

`Ext.` axis (axis `13`) is rotated so that in the background of captured garment is as least as possible disturbing objects. The best is single color flat surface. The `ext.` axis is stationary during the measurement.

3.3.2 Arm with Xtion Sensor

The record is captured with camera `xtion1` mounted on the arm `r1`. The arm `r1` moves into position where the optical axis of the camera heads horizontally. Simultaneously is the optical axis of the camera oriented towards arm `r2` (figure 3.1). The arm `r1` is stationary during the measurement.

3.3.3 Arm with Garment

Garment is held by gripper mounted on arm `r2`. The arm `r2` is in a height at which camera `xtion1` can capture motion of garment. The arm `r2` is in a position which it can perform motion required for the experiment (chap. 3.1). The arm `r2` makes motion so as garment moved perpendicularly to the optical axis. The arm `r2` makes a desired motion with the garment so that it rotates about an axis `B` certain angle and will return back to initial position. For better describe of the motion is motion mooted in the figure 3.2.

3.4 Capturing Background Image

This position is used for capture a reference image of background, for improve results of the experiment. We need background subtraction for processing RGB images. Background subtraction needs RGB image without garment on foreground. The captured image is used for filtering background (background substraction) from RGB image. More to background substraction is deal in the chapter 5.2.1. The turntable (`Ext.` axis) is rotated as in the case of capturing of RGB (chap. 3.3.1). The arm `r1` is into same position as in case of capturing of RGB (chap. 3.3.2). The reference image of background is captured that the arm `r2` (in which gripper is **not** held garment in this case) change position so that the arm `r2` is completely out of captured area of `xtion1`. In this position is performed the capture of background and the arm `r2` with the garment was returned to the position of measurement.

3.5 Capturing of Depth maps

3.5.1 Manipulator and sensor position

The turntable (`Ext.` axis) is rotated as in the case of capturing of RGB (chap. 3.3.1). The arm `r1` is into same position as in case of capturing of RGB (chap. 3.3.2).

3.5.2 Arm with Garment

Garment is held by the gripper mounted on arm `r2`. The arm `r2` is in a height at which camera `xtion1` can capture motion of garment. The arm `r2` is in a position which it can perform motion required for the experiment (chap. 3.1). The arm `r2` makes motion so as garment moved along to the optical axis. The arm `r2` makes a desired motion with the garment so that it rotates about an axis `R` certain angle and will return back to initial position. For better description of the motion is motion mooted in the figure 3.3.

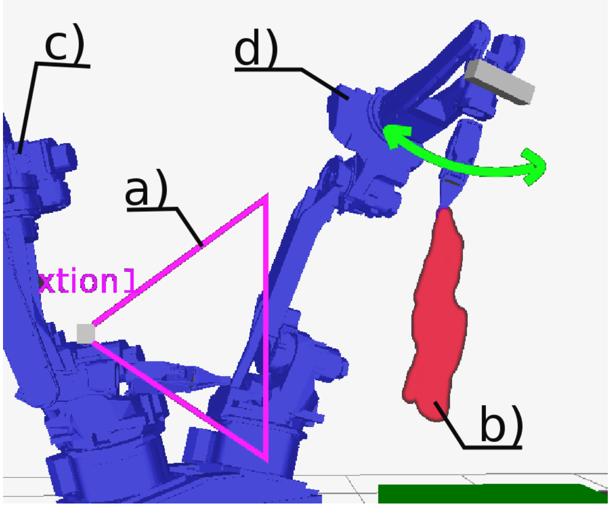


Figure 3.3. Suggestion of motions of gripper with garment along to optical axis. a) mooted field of vision (xtion1 sensor), b) garment, c) arm r1, d) arm r2.

Chapter 4
Data Structure

We decided to use offline processing for development of method and experiments. Therefore, it is important to store the measured data to data structure and then calculate the parameters that are important for the experiments (chap. 5).

4.1 Recording Data

Required data are stored using rosbag tool (chap. 2.2.1). Rosbag tool stores the data in the format `.bag` to the folder, which is set in the `local_options.py` file [1]).

The CloPeMa robot can produce over two hundred topics (chap. 2.2.1) when running. Due to the saving disk space and capacity of the transmission channel are recorded only topics which are important to the evaluation of the experiments. Selected topics are set in `topics.txt` [2]) and contains these choosen topics:

```
/joint_states
/tf
/xtion1/depth/camera_info
/xtion1/depth_registered/camera_info
/xtion1/rgb/camera_info
/xtion1/depth/image_raw
/xtion1/rgb/image_raw
/feedback_states
/r2_ee_to_depth
/r2_ee_to_rgb
```

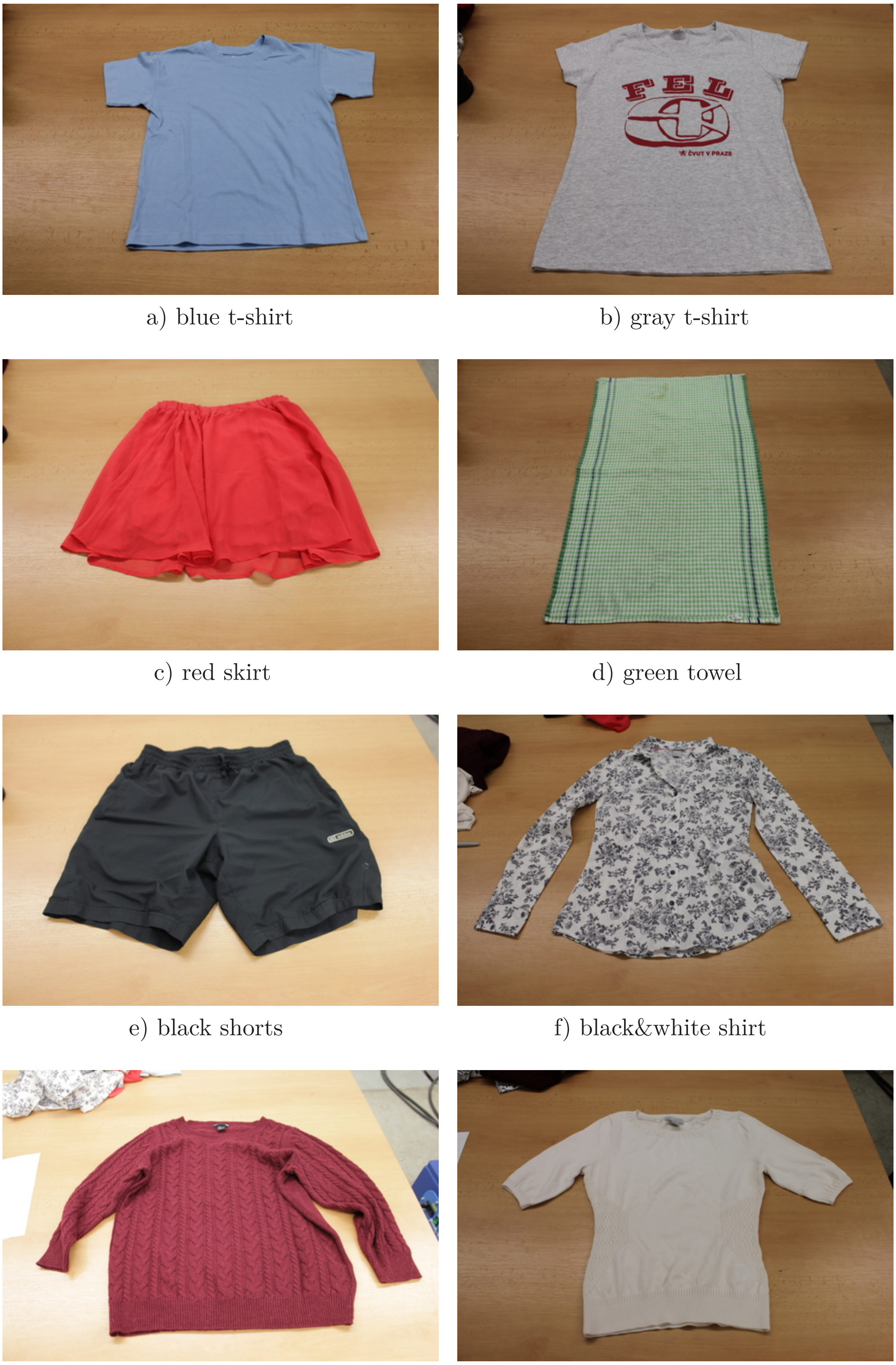
4.2 Measured Data Set of the Garments

4.2.1 Structure of Data Set

Within experiments was prepared a data set. The data set is used to validate the methods for estimation of the model parameters. The set contains garments with different shapes, colors, sizes and weights. The measured garments are described in the chapter 4.2.2 and are shown in the figure 4.1. The data set is stored on the CloPeMa project server.

[1]) `path_to_workspace/clopema_cvut/clopema_collect_model_data/src/local_options.py`

[2]) `path_to_workspace/clopema_cvut/clopema_collect_model_data/matlab/topics/topics.txt`



a) blue t-shirt

b) gray t-shirt

c) red skirt

d) green towel

e) black shorts

f) black&white shirt

g) red pullover

h) white pullover

Figure 4.1. The garments used in the experiments.

4.2.2 Description of the Garments

In the table 4.1 are described some parameters of the garment. The parameters were determined using external measuring instruments (scale, measuring tape). These parameters are used only as additional parameters for construct model of the garment and its verification. The additional parameters could be obtained by sensors of robot as well. Apart from the cases when is in the gripper held garment in the random point (in the figure C.1-a), are also captured special cases of holding garment in the gripper. One of this cases is holding the corner of the garment (in the figure C.1-b), collar (in the figure C.1-c) or belt (in the figure C.1-d) of the garment . Second special position is the position ”on the ruler” where the garment is straightened and outstretched (in the figure C.1-e and f). These special cases are only for simplify the simulating and virtual modeling of the garment at the beginning. The special cases are shown in the figure C.1.

type of the garment	color	size [cm]	weight [g]	picture
t-shirt	blue	47×58	72	a
t-shirt	gray	60×54	105	b
skirt	red	44×62	108	c
towel	green	60×36	45	d
shorts	black	50×60	173	e
shirt	black&white	69×120	137	f
pullover	red	65×120	446	g
pullover	white	54×76	184	h

Table 4.1. Description of the Garments. Last column refers to the figure 4.1.

4.2.3 Format of Names of Recorded Files

Each measurement of the garment produce two `rosbag` files. One with captured RGB images when garment is moving perpendicular to the optical axis, and one for captured depth maps when garment is moving along the optical axis. Recorded files are stored under different names accord to the form shows in the table 4.2.

name_speed_AX.bag	
name	choosen file name by user
speed	choosen speed of manipulator
A	axis, which was executed motion **R** or **B** (figure 2.3)
X	number of the topics file

Table 4.2. Explanation of format file name.

A robot operator selects a name of the file and speed of the moving garment. The name of the file stored in the prepared data set is chosen from type of the garment, color and name of the type of holding the garment. If the garment was measured and stored multiple times, the numeral was added. The list of the measured garments is described in the tables B.2 and B.3.

Chapter 5
Data Processing

5.1 Load Data to the MATLAB

The data are processed offline in MATLAB. The offline procesing is only for research purposes. We suppose that the methods will be implemented directly into the ROS and processed online in the future.

Data format `bagfile` are read using the `matlab_rosbag` tool (from [24]). The `matlab_rosbag` is a library for reading ROS bags in Matlab and it is licensed under the BSD license, making it suitable for use in CloPeMa project. The tool (library) can also read only selected topic, which is used in this case. The chosen topics are read from the same file [1]) which is used for definition of recording data (chap. 4.1). After loading the data into MATLAB, data are grouped into cells by topic. Next steps in the case of RGB images and depth maps are different.

5.2 Extraction of Features from RGB images

The sequence of RGB images is captured with the xtion camera (chap. 2.1.3). Images have resolution 640×480 px (width×height). RGB images are stored in the `bagfiles` as single row vector. After loading data into MATLAB (chap. 5.1) data are converted from row representation to three-dimensional RGB intensity representation $\mathbf{I}_{u,v}$ of images with dimensions 640×480 px ($m \times n$), where $\mathbf{I}_{u,v} = [\ r(u,v),\ \ g(u,v),\ \ b(u,v)\ ]$, $u \in \{1, 2, \dots, m\}$, $v \in \{1, 2, \dots, n\}$. $r(u,v)$, $g(u,v)$ and $b(u,v)$ are function of brightness of pixel on the coordinates u, v and each of $r(u,v)$, $g(u,v)$ and $b(u,v)$ represents relevant intensity of single color from the RGB color-space. Unless specified otherwise, the computations are always carried for each $r(u,v)$, $g(u,v)$ and $b(u,v)$ separately. Raw RGB data are shown in the figure 5.1.

[1]) `path_to_workspace/clopema_cvut/clopema_collect_model_data/matlab/topics/topics.txt`

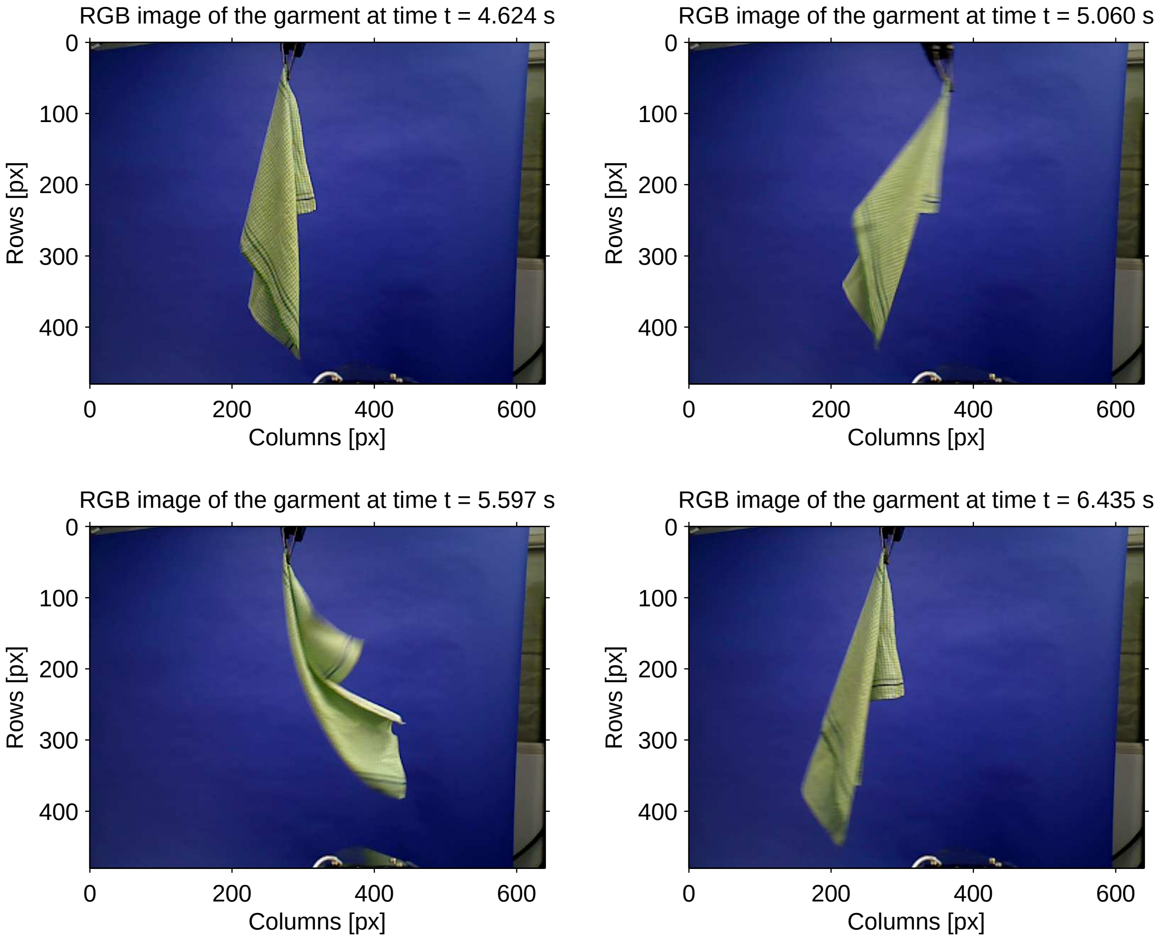

RGB image of the garment at time t = 4.624 s
RGB image of the garment at time t = 5.060 s
RGB image of the garment at time t = 5.597 s
RGB image of the garment at time t = 6.435 s
Rows [px]
0
100
200
300
400
Columns [px]
0
200
400
600

Figure 5.1. RAW RGB images.

5.2.1 Background Subtraction

In the chapter 3.4 is described a way of capturing background image. Essentially, was not captured a single image, but was captured a sequence of N images $\mathbf{F}^i = \{\mathbf{F}^1, \mathbf{F}^2, \ldots, \mathbf{F}^N\}$ in the time $i \in \{1, 2, \ldots, N\}$, where $\mathbf{F}^i_{u,v} = [r(u,v),\ g(u,v),\ b(u,v)\ ]$. The sequence of background images $\mathbf{F}^i_{u,v}$ is averaged to $\overline{\mathbf{F}}_{u,v} = [r(u,v),\ g(u,v),\ b(u,v)]$ according to (5.1). Together with mean value $\overline{\mathbf{F}}_{u,v}$ of RGB background images, is computed corrected sample standard deviation $\sigma_{u,v} = [\ \sigma_R(u,v),\ \sigma_G(u,v),\ \sigma_B(u,v)\ ]$ (5.2), whith is used as a intensity threshold.

$$\forall u, v : \overline{\mathbf{F}}_{u,v} = \sum_{i=1}^{N} \frac{\mathbf{F}^i_{u,v}}{N} \tag{5.1}$$

$$\forall u, v : \sigma_{u,v} = \sqrt{\frac{1}{N-1} \sum_{i=1}^{N} \left(\mathbf{F}^i_{u,v} - \overline{\mathbf{F}}_{u,v}\right)^2} \tag{5.2}$$

From the RGB image $\mathbf{I}$ is computed a binary silhouette image $s(u,v)$ according to (5.3),where $\alpha > 1$ is a threshold coeficient.

$$\forall u,v : s(u,v) = \begin{cases} 1, & \text{if} \begin{cases} & \left|r(u,v) - \overline{\mathbf{F}}_R(u,v)\right| > \alpha\,\sigma_R(u,v); \\ \text{or} & \left|g(u,v) - \overline{\mathbf{F}}_G(u,v)\right| > \alpha\,\sigma_G(u,v); \\ \text{or} & \left|b(u,v) - \overline{\mathbf{F}}_B(u,v)\right| > \alpha\,\sigma_B(u,v); \end{cases} \\ 0, & \text{otherwise.} \end{cases} \tag{5.3}$$

Silhouette images are shown in the figure 5.2.

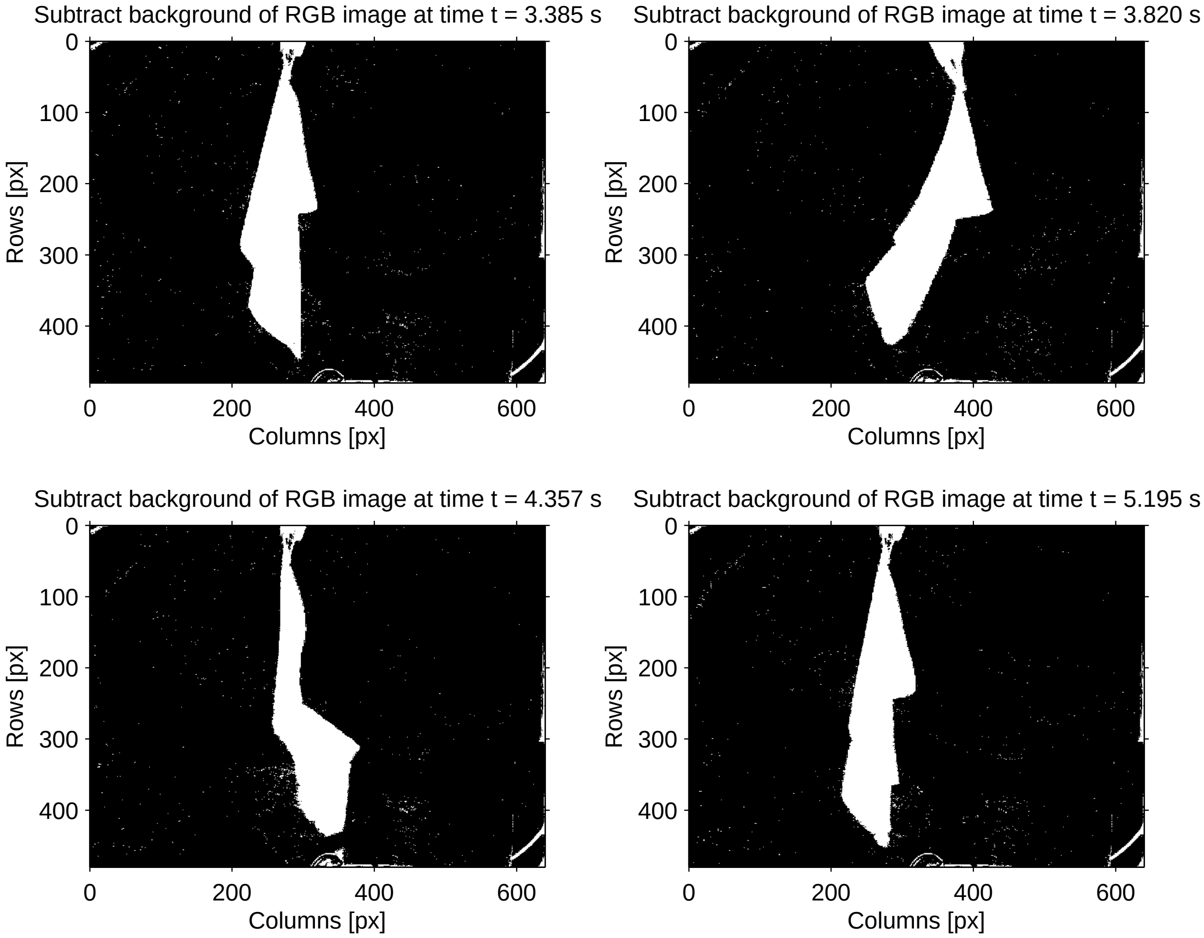


Figure 5.2. Binary silhouette images - background subtraction.

The figure 5.2 shows that background subtraction is not perfect. This imperfection has two main reasons. First reason is that the the gripper of arm `r1` and the surrounding of monochrome background are partly seen in the image. This would not be a problem for our method, if these objects have not a shiny finish, which reflects the garment when the garment is moving. By experimental results, we found that the garment moves only against a monochrome background and not enter the background surroundings during the entire motion. This imperfection is removed with help of a clipping mask, which cover gripper and monochrome background surrounding.

Second reason is a noise on the RGB image and tiny changes of brightness of pictures, which is caused by the garment motion. This imperfection is removed with help of morphological operation **opening**. Because this imperfection may occur in reverse too, i.e. so that the point of the garment is detected as a background thus **closing** operation followed opening operation (mean of these morpholigical operation is explained in [25] p.667-669). We use for this operations the MATLAB functions `imopen()` and `imclose()`.

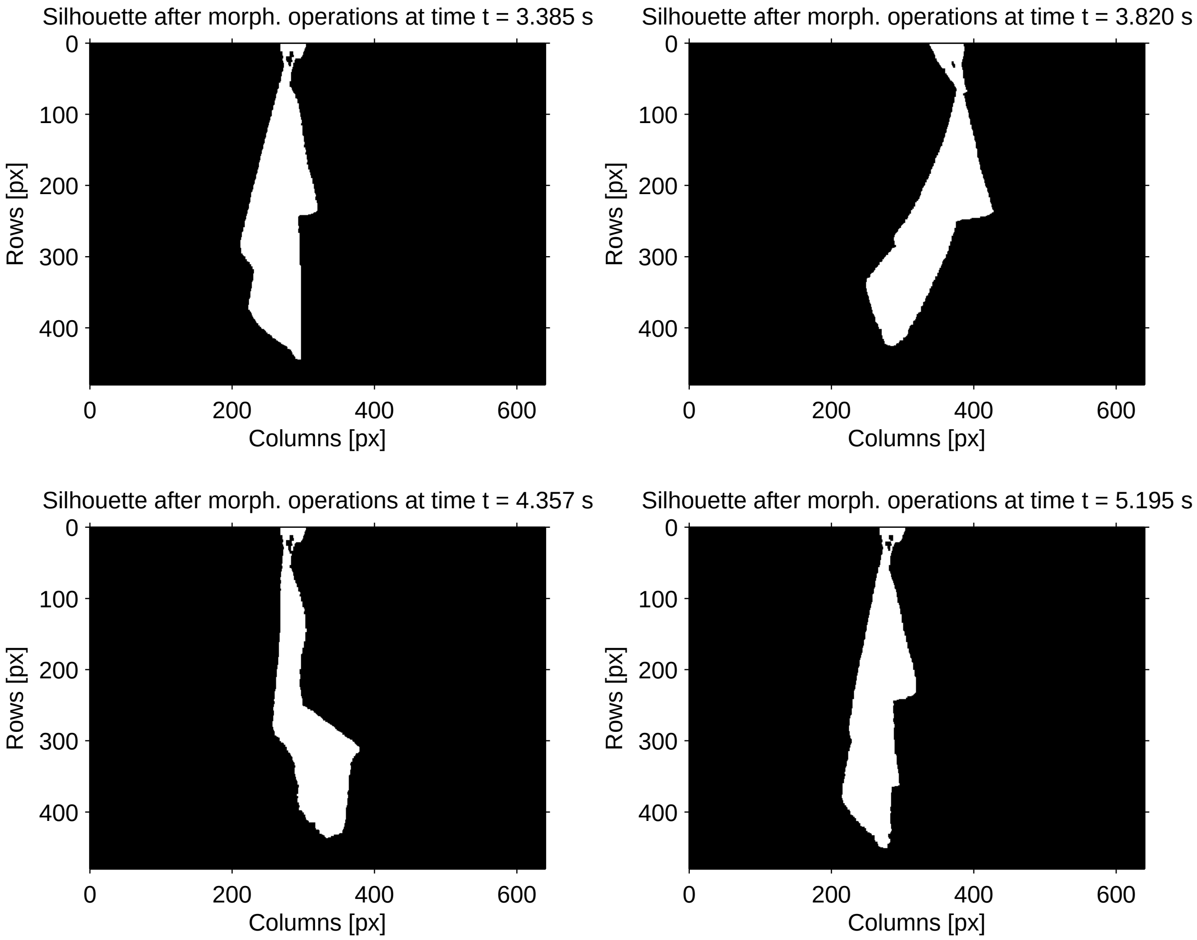


Figure 5.3. Background subtraction of RGB images after morphological operation.

5.2.2 Finding Major Axis of the Garment and Remove Gripper

After applying the previous steps we can assume that the garment is represented by the largest connected region in the image. Therefore, we focuse on the largest region. We use MATLAB functions `bwconncomp()`, which finds all objects in an image. For these object is individually calculated their size. Subsequently, according to these sizes determine the largest object and take out the rest. The detailed explanation of these region identification is described in the [25] (p.332).

In the next step we found major and minor axis of the garment. Befere we can find these parameters we have to remove from the image a showed end of the gripper. We have captured images without measured garment in the gripper, thus we can find an area in which the gripper is. Then this part of the image is removed from the image of the measured garment. A better way would be to compare the captured image data with a simulated robot model in an environment of ROS.

After the garment was removed from the gripper, the major and minor axis could be found by using the spatial moments (described in [26] p.74,658). This method describes finding major and minor axis for an ellipse (reasons are described below in the chap. 5.2.3). The axes are found using MATLAB functions `regionprops()` which gave us center of the object, lenght of major and minor axis and clockwise orientation of the major axis. Found major and minor axes of the silhouette of the garment and subtraced images without the gripper are shown in the figure 5.2.

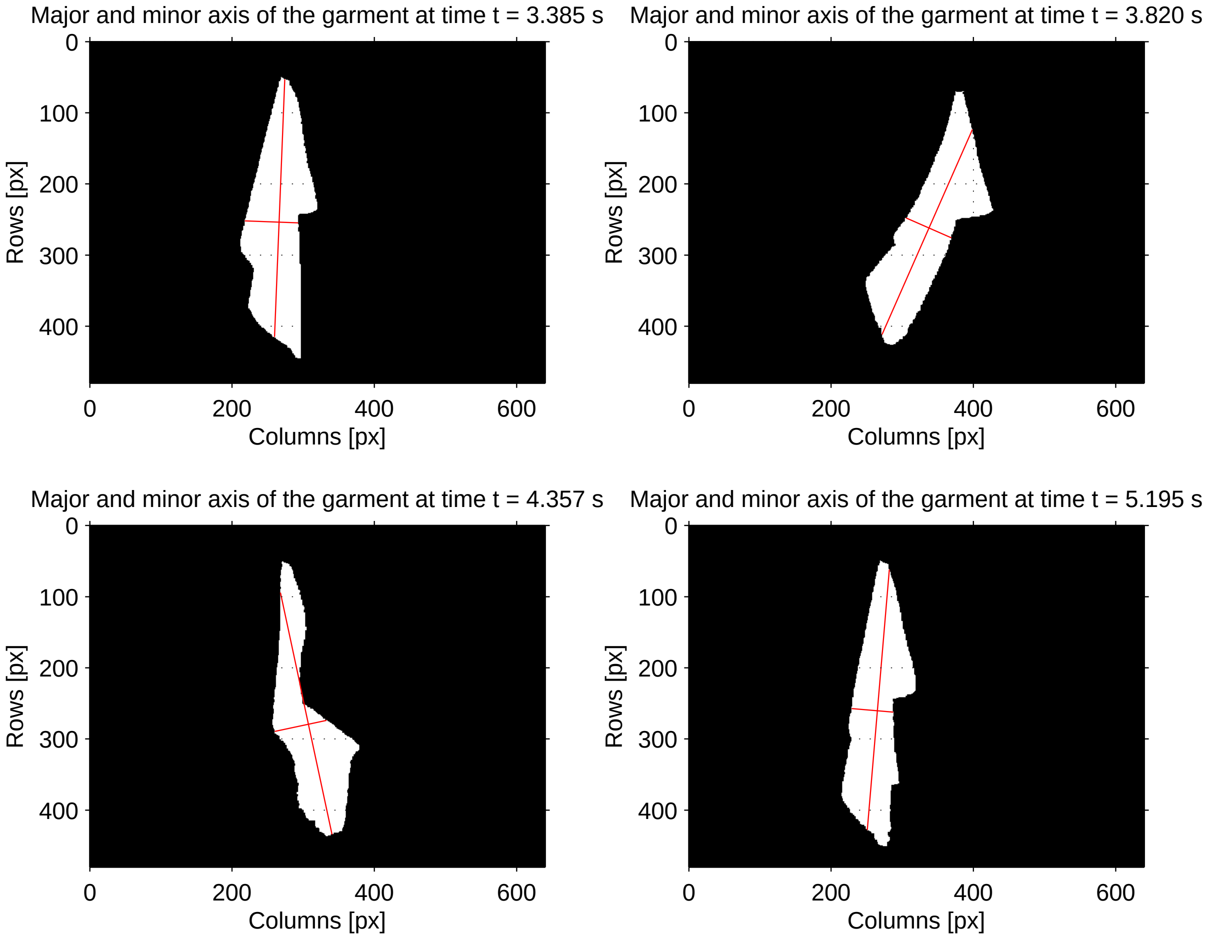


Figure 5.4. Major and minor axes of the silhouette of the garment and subtraced images without the gripper.

5.2.3 Finding Central Curve of the Garment

To obtain the garment position, we decided to track a central curve of the silhouette. During this project we have tried various methods of finding the curve, for example, the average pixel with respect to the rows of image region, or find a skeleton of the silhouette image ([25], p.365). Searching a skeleton of the silhouette image seemed as the most promising at the beginning, but then there was a problem in the form of determining the curve at the ends of the garment.

Finally, we chose finding of the central curve so the search for the centers of the corresponding cuts, which are perpendicular to the major axis of the silhouette.

The first step to obtain the central curve of the silhouette is finding the outline of the silhouette. I used the another property of the MATLAB function `regionprops()`, which contains an edge pixels of the garment. Edge of the silhouette is shown in the figure 5.5. For the next processing is important to convert pixels coordinating the edge of the garment to the border curve. For this conversion we use inner boundary tracing algorithm (which is described in the [25] p.191). In our case, the MATLAB function `bwboundaries()` is used as the implementation of this algoritm.

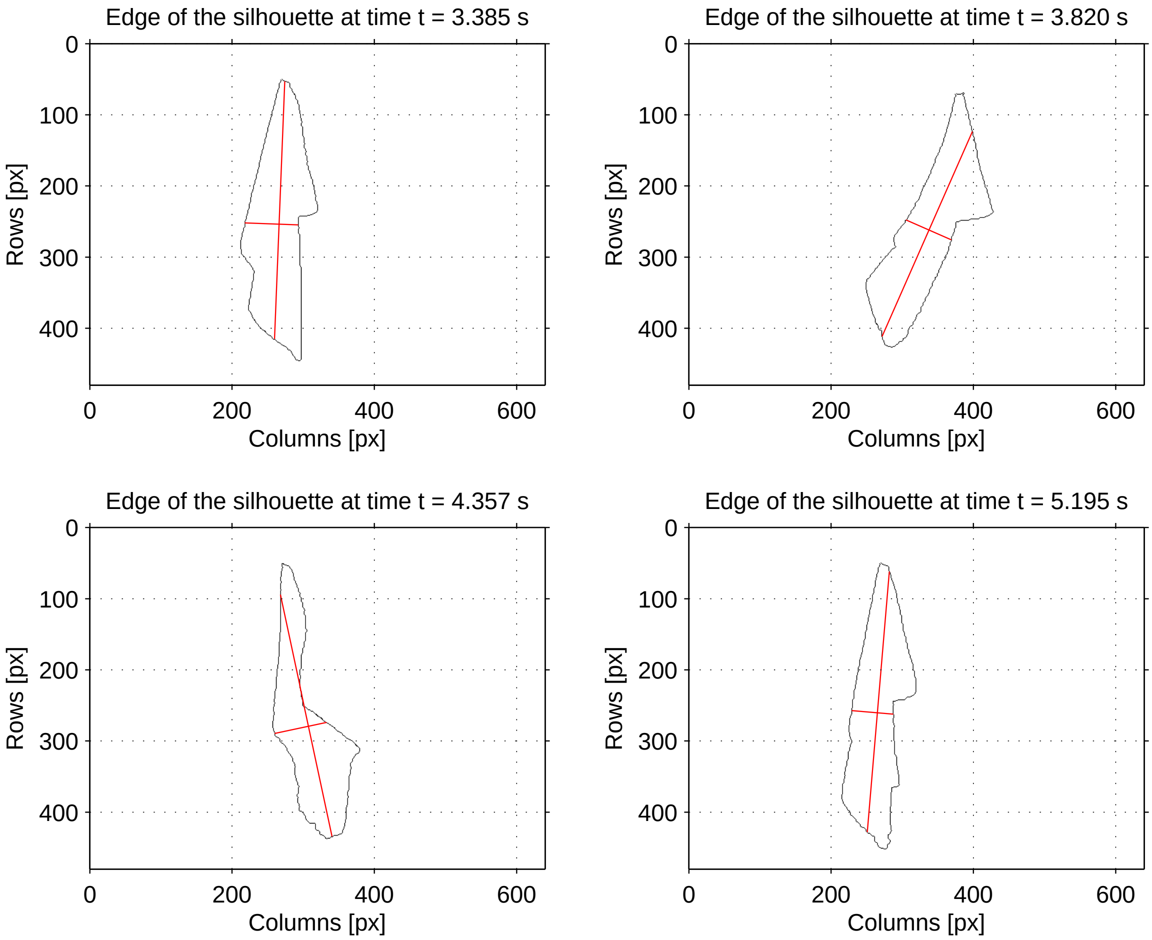


Figure 5.5. Edge of the silhouette of the garment.

The final step is to find the central curve. The point of the central curve is founded with help of the corresponding cut line as it is described earlier in this chapter. The center of the silhouette is represented like midpoint of the points, which are computed as an intersections of the border curve and corresponding cut line. For computing this curve intersections are used as the fast and robust curve intersections algorithm [27]. Found center curve of the silhouette is shown in the figure 5.6.

5.2.4 Finding Mathematical Features from RGB images

To estimate the parameters of the garment, we wanted to use the points of the central curve of the silhouette over time. Unfortunately, the position of the points forming the central curve is not influenced only by observed motion of the garment, initialized by the robot. The silhouette and the central curve is influenced by the complex motion of the garment. It is unsuitable for basic experiments. We decided not to use RGB images to estimate the parameters of dynamic physical model of the garment.

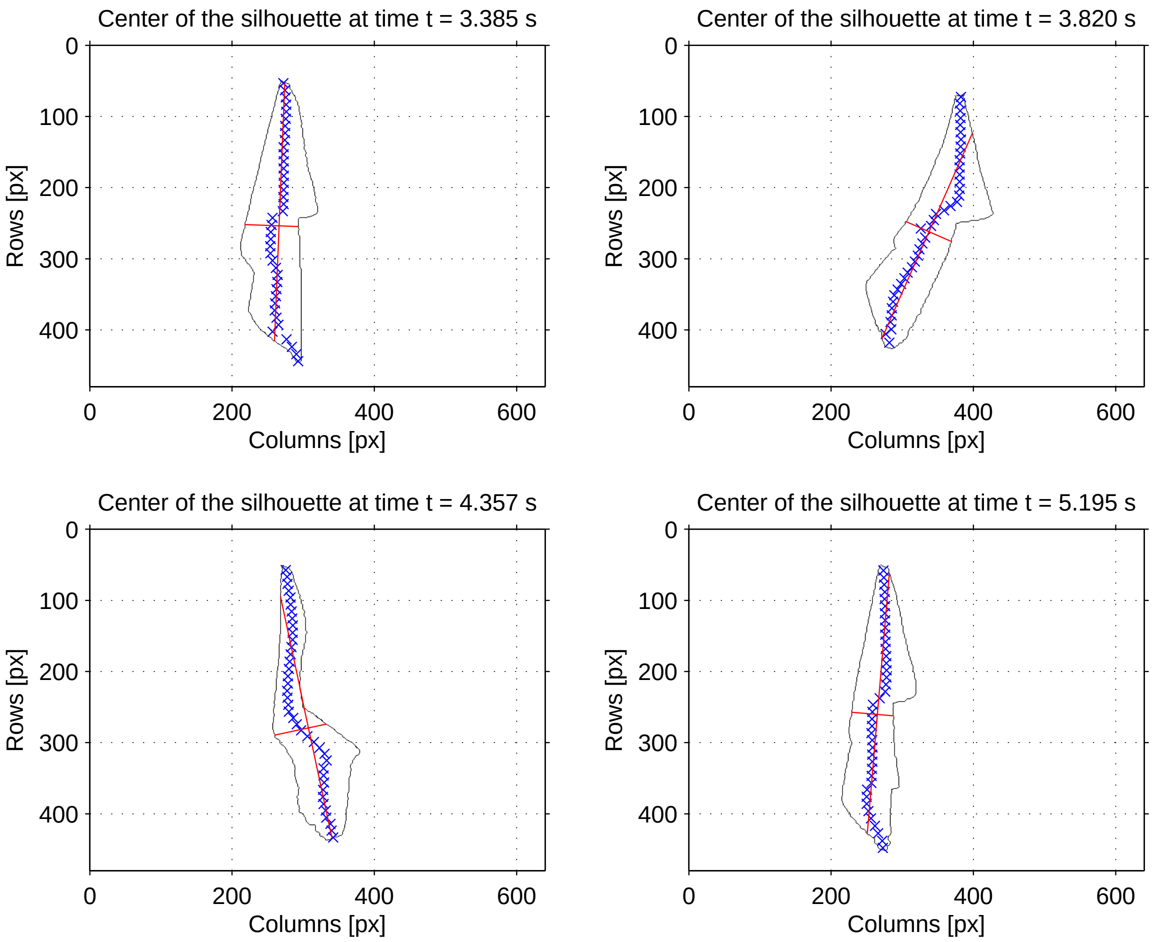


Figure 5.6. Found center curve of the silhouette.

5.3 Extraction of Features from Depth Map

The sequence of depth maps is captured with the Xtion camera (chap. 2.1.3). Depth maps are in a sufficient resolution 640×480 px (width×height). Depth maps are stored in the `bagfiles` as single row vector. After loading the data into MATLAB (chap. 5.1) data are converted from row representation in milimeters to depth map $\mathrm{C}(u, v) = z$, where z is a distance from the Xtion camera in meters, u and v are the depth map image coordinates ($u \in \{1, 2, \ldots, n\}; v \in \{1, 2, \ldots, m\}$). The depth maps are shown figure 5.7.

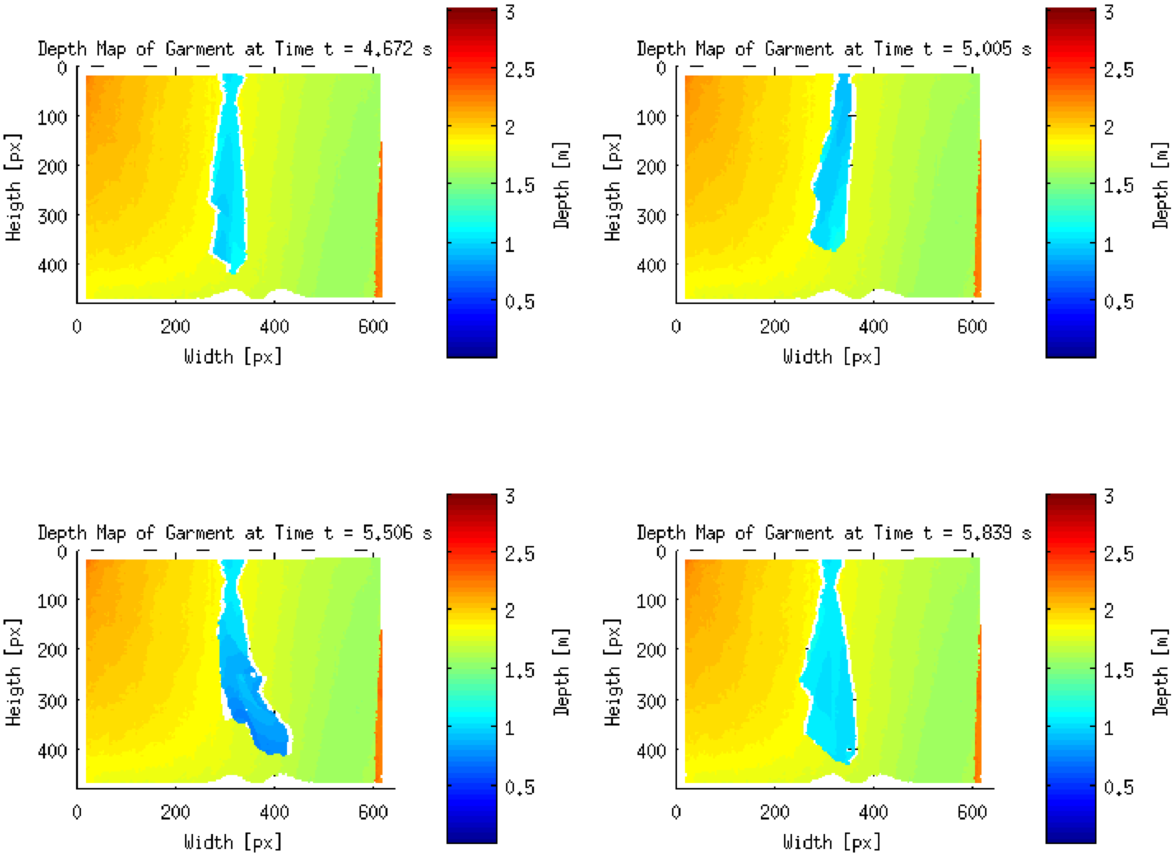


Figure 5.7. Depth Map obtained by Xtion rangefinder.

5.3.1 Convert Depth Map to 3D points

Together with data depth map is also stored calibration parameters of the Xtion rangefinder. From these calibration parameters comes calibration matrix **K** (eq. (5.4)), where f_x and f_y are partial focal lengths and (c_x, c_y) is a principal point.

$$\mathbf{K} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \tag{5.4}$$

Using the calibration matrix **K** can be depth map $\mathrm{C}(u, v)$ converted into 3D point cloud (set of points) **M** (eq. (5.5)), where $\mathbf{M}^i = [x^i, y^i, z^i]$ is point represent by its euclidean coordinates. We use usually coordiate system of the Xtion sensor.

$$\mathbf{M} = \{\mathbf{M}^1, \mathbf{M}^2, \ldots, \mathbf{M}^{m \times n}\} \tag{5.5}$$

The point cloud $\mathbf{M}$ is constructed according to equation (5.6). Algorithm used for these conversion was inspired by [28]. We obtain each point $\mathbf{M}^i \in \mathbf{M}$ from depth map $\mathrm{C}(u, v)$ by:

$$\begin{aligned} z^i &= \mathrm{C}(u, v) \\ x^i &= \frac{z \cdot (u - c_x)}{f_x} \\ y^i &= \frac{z \cdot (v - c_y)}{f_y} \end{aligned} \tag{5.6}$$

where $u \in \{1, 2, \ldots, n\}$, $v \in \{1, 2, \ldots, m\}$ and $i \in \{1, 2, \ldots, n \times m\}$. Example of data from 3D points are shown in the figure 5.8.

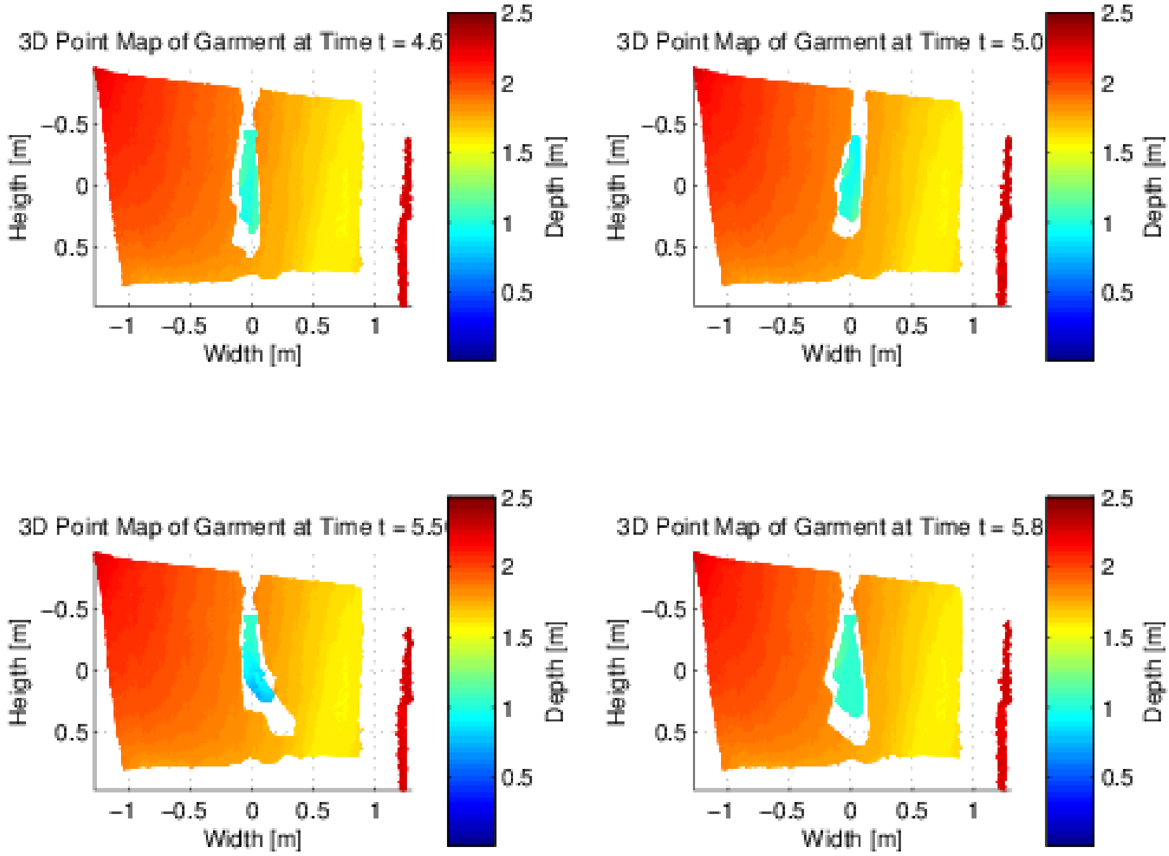


Figure 5.8. Point cloud of the garment in different times.

5.3.2 End of the Gripper

In the figure 5.11 are not shown only points on the garment. The figure shows points of the gripper of arm `r2` 2.1.3 too. As has been mentioned in the chapter 3.5 the motion of the garment is caused by this gripper. Localization of the coordinates **ee** $= [ee_x, ee_y, ee_z]$ of the end point of the gripper is important not only for resoluting the gripper from the garment, but also for detecting excitation motion of garment.

The coordinates are found by information about robot position. The simple service in ROS was written for this operation (chap. 2.2.1). This service subscribes to the topic of `/xtion1/depth/image_raw`. This topic generates a message each time an depth map image is captured. For every incoming message is computed a coordinates of **ee** — end of the gripper with garment. To obtain these coordinates is used `lookupTransform()` function from the ROS environment. The function uses the `tf` package of ROS which contains the relationships between parts of manipulator in a tree structure and allows to obtain a transformation between any two portions of the manipulator [29]. This coordinates are not in the global coordinate system, but in the coordiate system of `xtion1` camera. Then, the service publishes data as a new message in its own topic. So the position of the end point of the gripper is captured for every depth map image. Located gripper is shown in the figure 5.9.

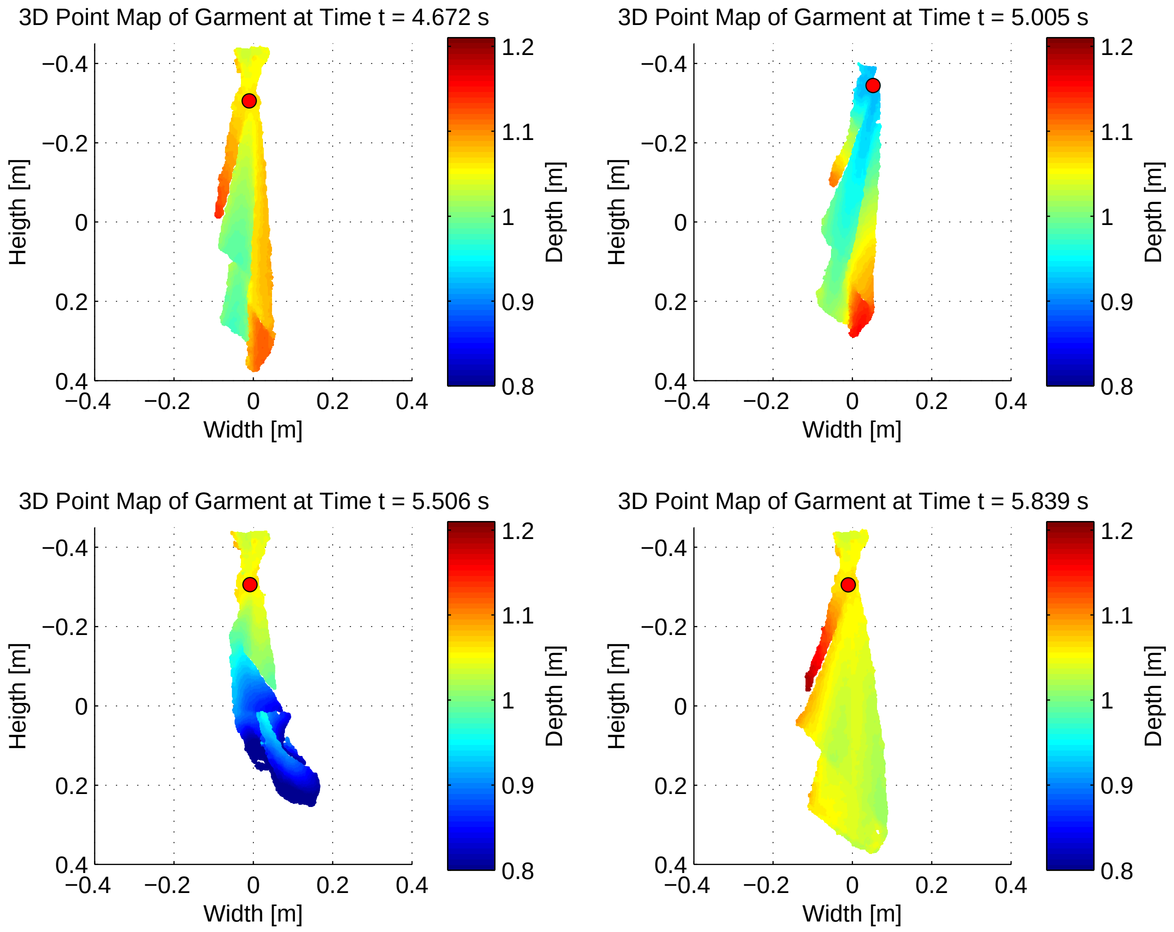


Figure 5.9. Filtred points of measured object with the gripper (gripper position shows a red circle).

5.3.3 Filtering by Depth of Area

The figure 5.8 shows that the Xtion camera captured not only the points on the garment but also captured the unnecessary surroundings. We decided filtering the 3D points according to distance from Xtion camera. We know that the gripper of the arm `r2` with garment is located 1.06–1.07 m before the Xtion camera on the arm `r1` at its default measurement position. After several experiments, we found that the minimum and maximum of z-axis values on the garment have values between 0.8 m to 1.16 m, thus we know that the garment moves between these values.

If the deviation is known, point cloud $\mathbf{M}$ (from (5.6)) can be filtered to point cloud $\mathbf{M}^*$ (eq. (5.7)), to extract the garment shape (according to eq. (5.8)).

$$\mathbf{M}^* = [x^*, y^*, z^*] \tag{5.7}$$

The filted data are shown in the figure 5.10. You can see the differences between figures 5.7 and 5.10. Moreover, this method of "background substraction" is much robust than method of background substraction of RGB images 5.2.1.

$$\begin{gathered} \mathbf{M}^* \subseteq \mathbf{M} \\ \mathbf{M}^* = \{\mathbf{M}^i = [x^i, y^i, z^i];\, ee_z - \Delta z \leq z^i \leq ee_z + \Delta z\} \end{gathered} \tag{5.8}$$

when Δz is range of motion of the garment according to the gripper coordinates. We chose a value of Δz three times to value determine on the base of experimental results.

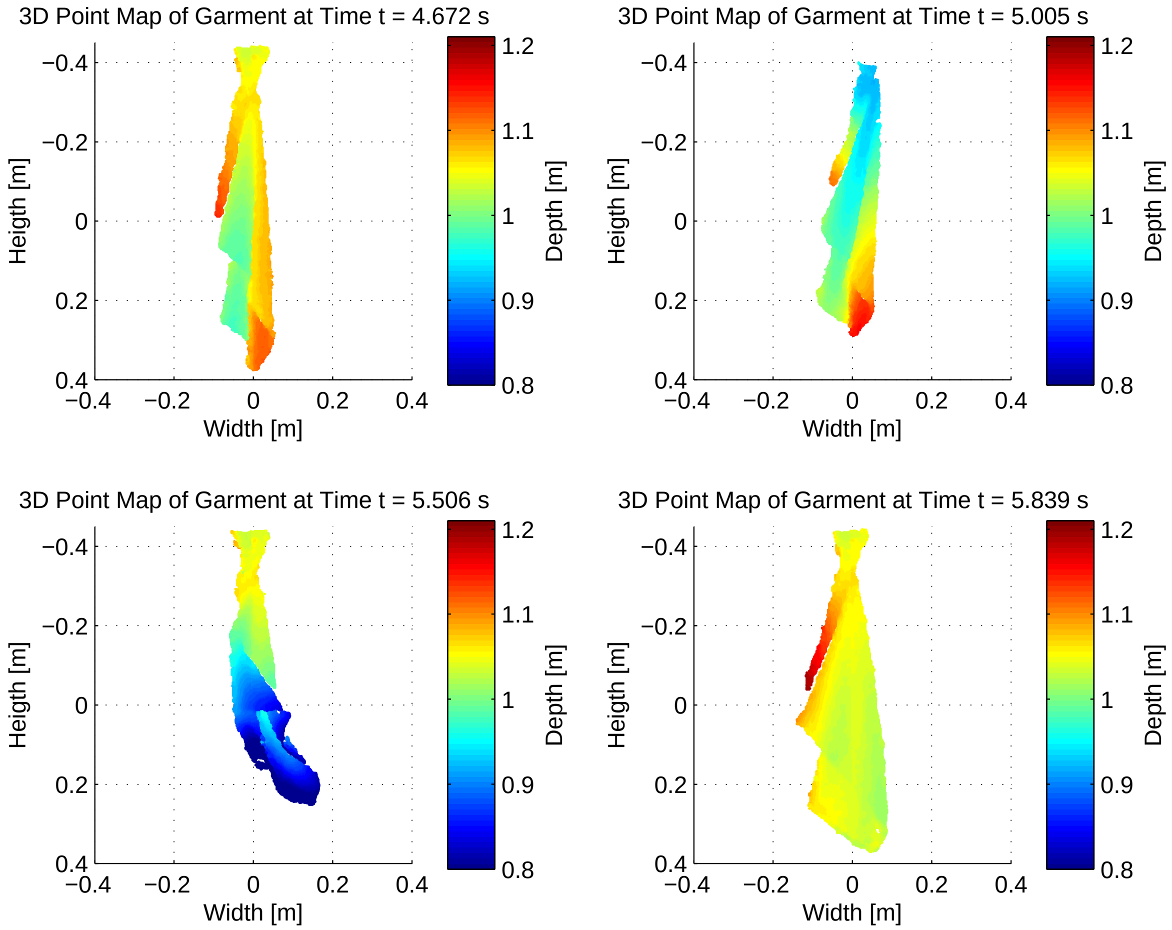


Figure 5.10. Filtred point cloud of a garment.

5.3.4 Finding Points

First of all is reduced the 3D points matrix. Points over the gripper endpoint **ee** (from chap. 5.3.2) are not needed anymore. The points are filtred by the y-axis from point cloud $\mathbf{M}^*$ to the point cloud $\mathbf{G}$.

We suppose a simple model usefull for parametric estimation. The model can be imagined as a chain of connected segments pendulous vertically down. We need to obtain the position of points on the garment vertically down from the gripper. Vertical direction will be denoted as the direction parallel to the y-axis.

Because the motion is limited (described in the chap. 3.2), we can assume that the point cloud $\mathbf{G}$ represents a some function $z = \mathrm{f}(y)$, where y is the vertical coordiate of the point on the garment and for all z will be the x^i constant and answer to the end of the gripper position. This function can be obtained from the point cloud by sampling. Sampling is chosen in the vertical direction down from gripper position. Sampling intervals are chosen based on a raster which provides a sensor. The raster distances are approximately (on our dataset) $\Delta y_{us} = 2.5\ mm$. So we chose the sample region size $\Delta y = 5\ mm$, to which we almost always received at least one point in each sample. We selected coordinates y^i with equidistant step Δy_{us}. We also set the sample size Δx, between the points and vertical line, which is parallel to the y-axis and intersects end of the gripper. We found sets $\mathbf{L}^i$, where $i \in \{1, 2, \ldots, k\}$ and k is number of sampling intervals vertically down from the gripper to end of the garment. Let $\mathbf{L}^i = \{\mathbf{p}^i_1, \mathbf{p}^i_2, \ldots, \mathbf{p}^i_o\}$ is set of points, where o is number of points contains in $\mathbf{L}^i$, then the points $\mathbf{p}^i_j$ ($\mathbf{p}^i_j \in \mathbf{G}$) meet the following conditions:

$$\begin{aligned} \mathbf{p}^i_j &= [x^i_j, y^i_j, z^i_j] \\ \forall i, j : (x^i - \Delta x) &< x^i_j < (x^i + \Delta x) \\ \forall i, j : (y^i - \Delta y) &< y^i_j < (y^i + \Delta y) \end{aligned} \tag{5.9}$$

Then central point $\mathbf{s}^i = [x^i, y^i, z^i]$ of set $\mathbf{L}^i$, where we know y^i, is determined as:

$$\mathbf{s}^i = \left[x^i = \frac{1}{o}\sum_{j=1}^{o} x^i_j, y^i, z^i = \frac{1}{o}\sum_{j=1}^{o} z^i_j \right] \tag{5.10}$$

where o is the number of elements of $\mathbf{L}^i$. According to the equation (5.10) we found the chain of the sample points $\mathbf{s}^i$, where $\mathbf{s}^i$ is sorted according to y^i.

In the thesis [30] was made the requirement for location of measurement equidistant points $\mathbf{P} = \{\mathbf{P}_1, \mathbf{P}_2, \ldots, \mathbf{P}_r\}$ on the garment, where $\mathbf{P}_q = [x_q, y_q, z_q]$ and r is number of required points. According to the require points are located in the middle of equally long segments and value of require points r is five in a presented experiment (but it is changeable). Therefore, the next step is finding points based on constant length of the garment.

First step is found lenght of the garment. The length l of garment is summarized according to sample points $\mathbf{s}_i$ (eq. (5.11)).

$$\forall \mathbf{s}_i : l = \sum_{i=1}^{j-1} ||\mathbf{s}_i - \mathbf{s}_{i+1}|| \tag{5.11}$$

where j is the number of elements of $\mathbf{s}$. The length l of the garment is divided by the number of required points. With this dividing are made equally long segments. The required points **P** are in the middle of these segments. These points (with time stamps) are forwarded for further processing. Found points are shown in the figure 5.11.

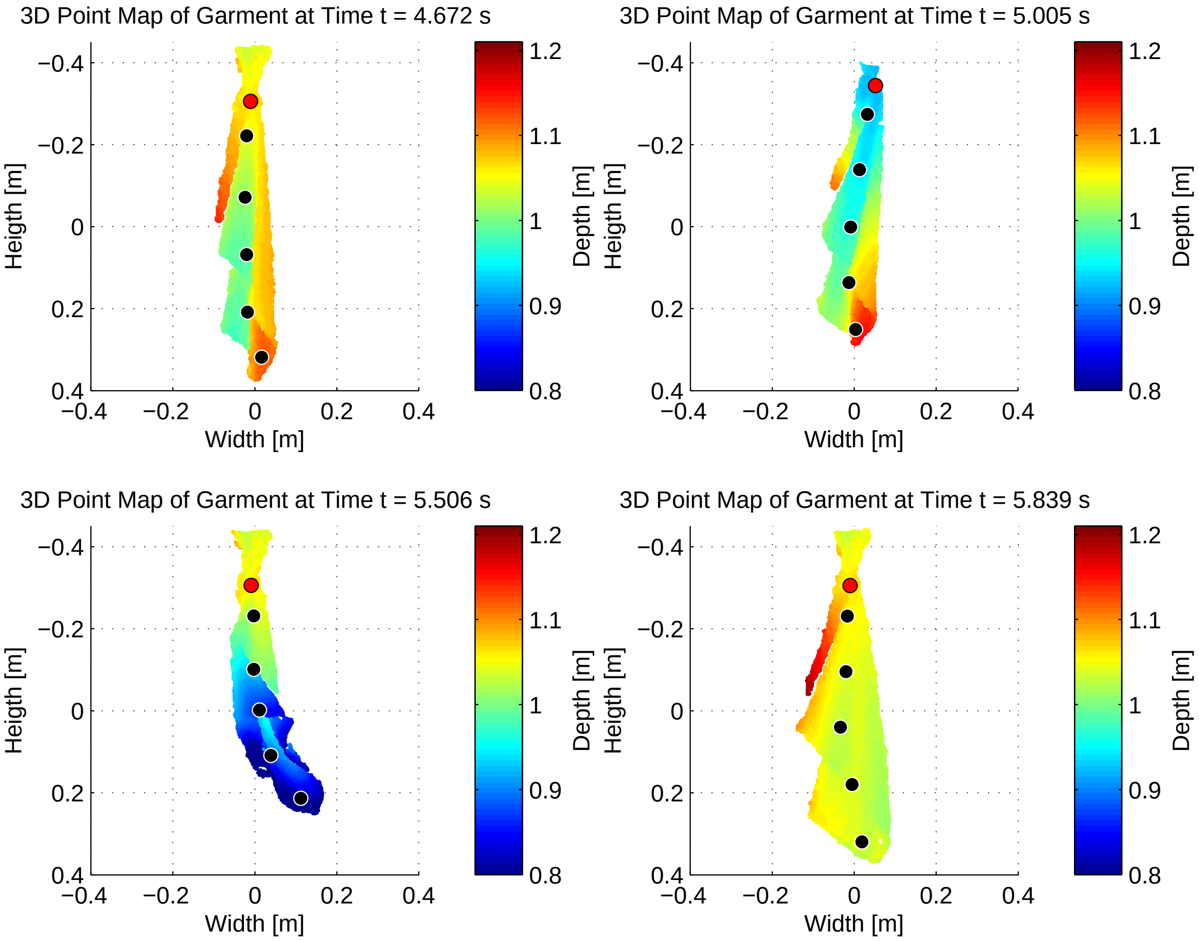


Figure 5.11. 3D point cloud with points found on the garment.

5.3.5 Finding Mathematical Features from Depth Map

Points **P** (from chap. 5.3.4) are obtained from whole sequence of the measurement of the garment. Because we found points **P** including timestamps, we can examine the dependence of displacement on time or dependence of speed on time. The timestamps for each depth map we obtained also from rosbag file.

The dependence of displacement Δz_k on the time is shown in the figure 5.12 and dependence of speed of the points on the time is shown in the figure 5.13. Figures show found points **P** (blue) and also they show position or speed of the end of the gripper **ee**. Both figures show specified dependences to green towel captured in the special case of holding of the garment straightened on the ruler.

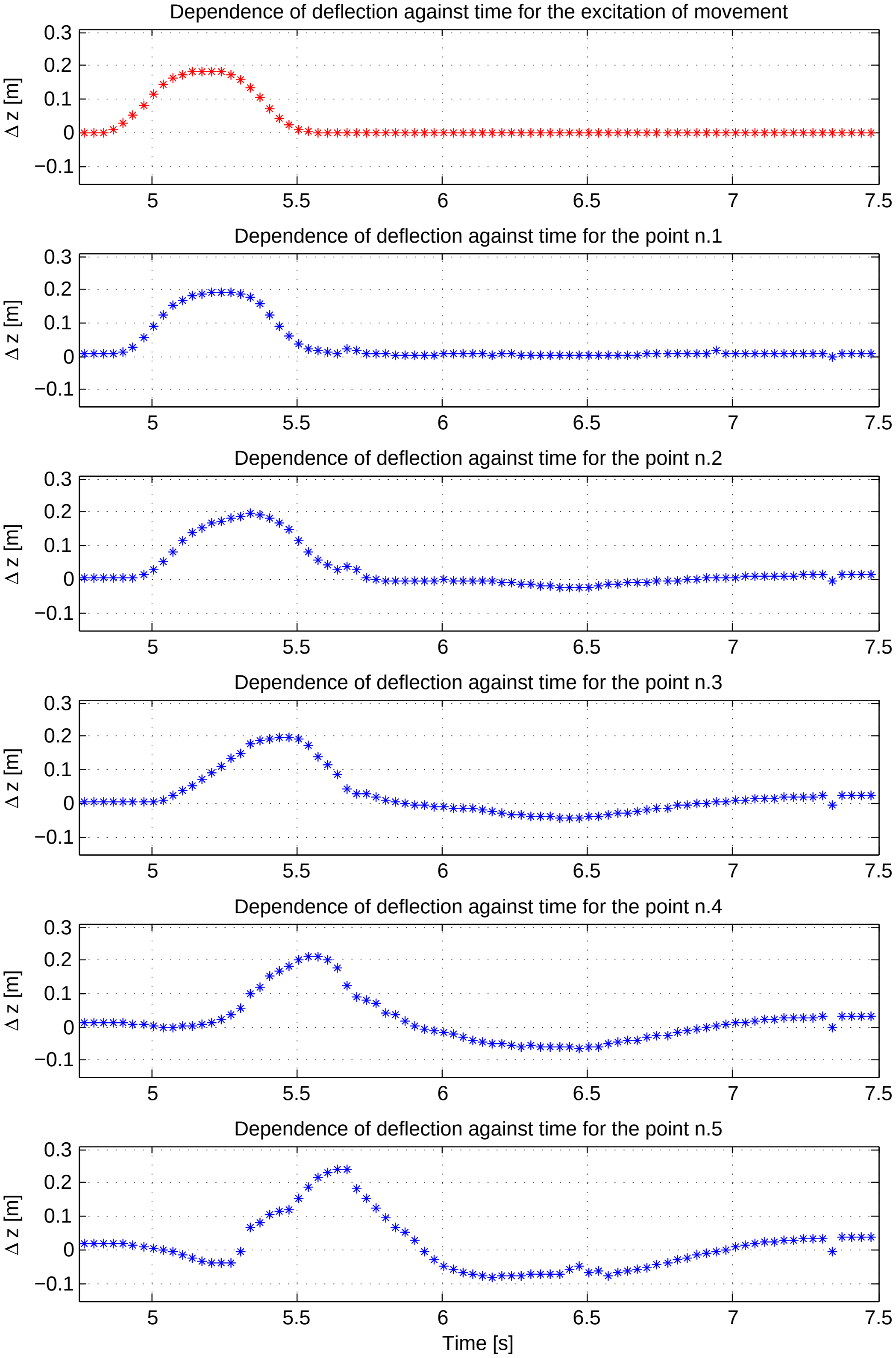


Figure 5.12. Motion of the garment, dependence of the position of points on the time.

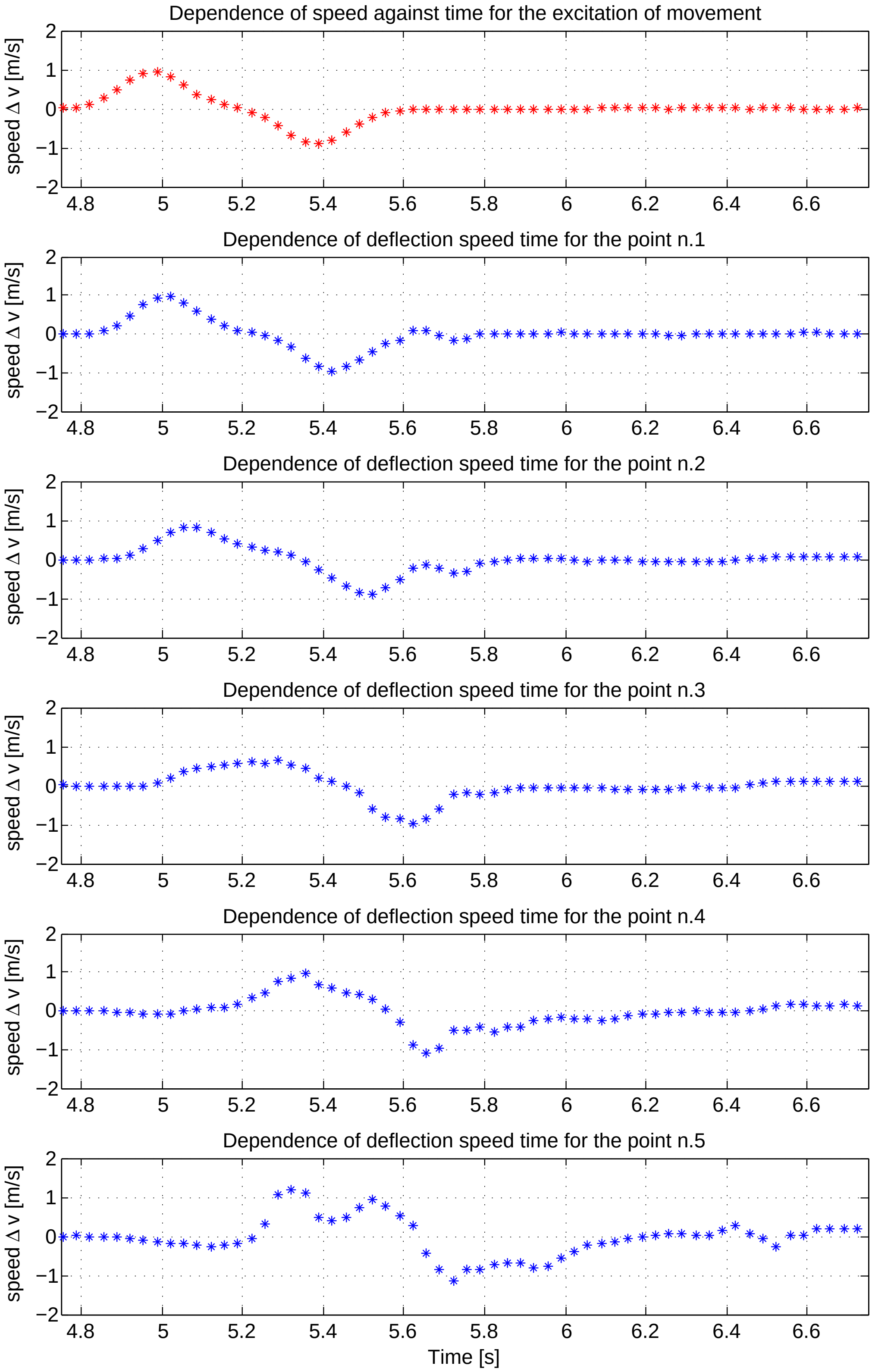


Figure 5.13. Motion of the garment, dependence of the speed of points on the time.

Chapter 6
Results and Discussion

One of the results of our work is the data set. The data set contains forty-six experiments with eight different garments (described in the 4.2.1), which are held in the gripper in five different ways (described in the appendix C). The garments are described in the appendix B. Overall, this data set represents 15 gigabytes of data. This is due to the fact that for each motion are captured RGB video and depth maps simultaneously at full resolution and additional information about the position of the robot. The large data is not part of enclosed DVD. So the enclosed DVD contains only selected examples of the data. Selected examples are labeled in the appendix B.

The figures 5.12 and 5.13 are typical measurement from the dataset, it shows how the garment moves. In the referenced example is shown the garment called green towel (shown in the figure 4.1.d) which is held in its corner. Data in this example come from the depth map sensor. You can see that the motion of the closest point of the gripper is very similar to move of the gripper. Other points, further from the gripper, are move with time shift. The further the garment is from the gripper the more its motion resembles natural frequencies of the hanging garment. The shape of the time course of changes, which may be declared to the expected result.

For the documentation repeatability of the results of experiments I chose this experiment: In the case of the garment - green towel - I made five experiments repeatedly, so I can compare if the behavior is stable under small changes in the experiment. The figure 6.1 indicate the relative position to the static condition over time and in the figure 6.2 is shows the speed of individual points. The length of hanging garment was approximately 64 cm. The graph shows the 5 points and gripper that creates the motion. The first point is in the idle state 6.4 cm below the gripper and other points are placed sequentially down from the gripper at intervals 12.8 cm.

Into the graphs are always plotted the results of all five experiments. The measurement was carried out so that after the end of one experiment, the garment was allowed to stabilize and then the experiment was performed again. It is seen that the waveforms of these five experiments are very similar. Only points that are farthest from the gripper have slight variations, which is due to the expected behavior of the garment. A same behavior can be seen in both graps, position and speed. Such a similarity of waveforms we expected.

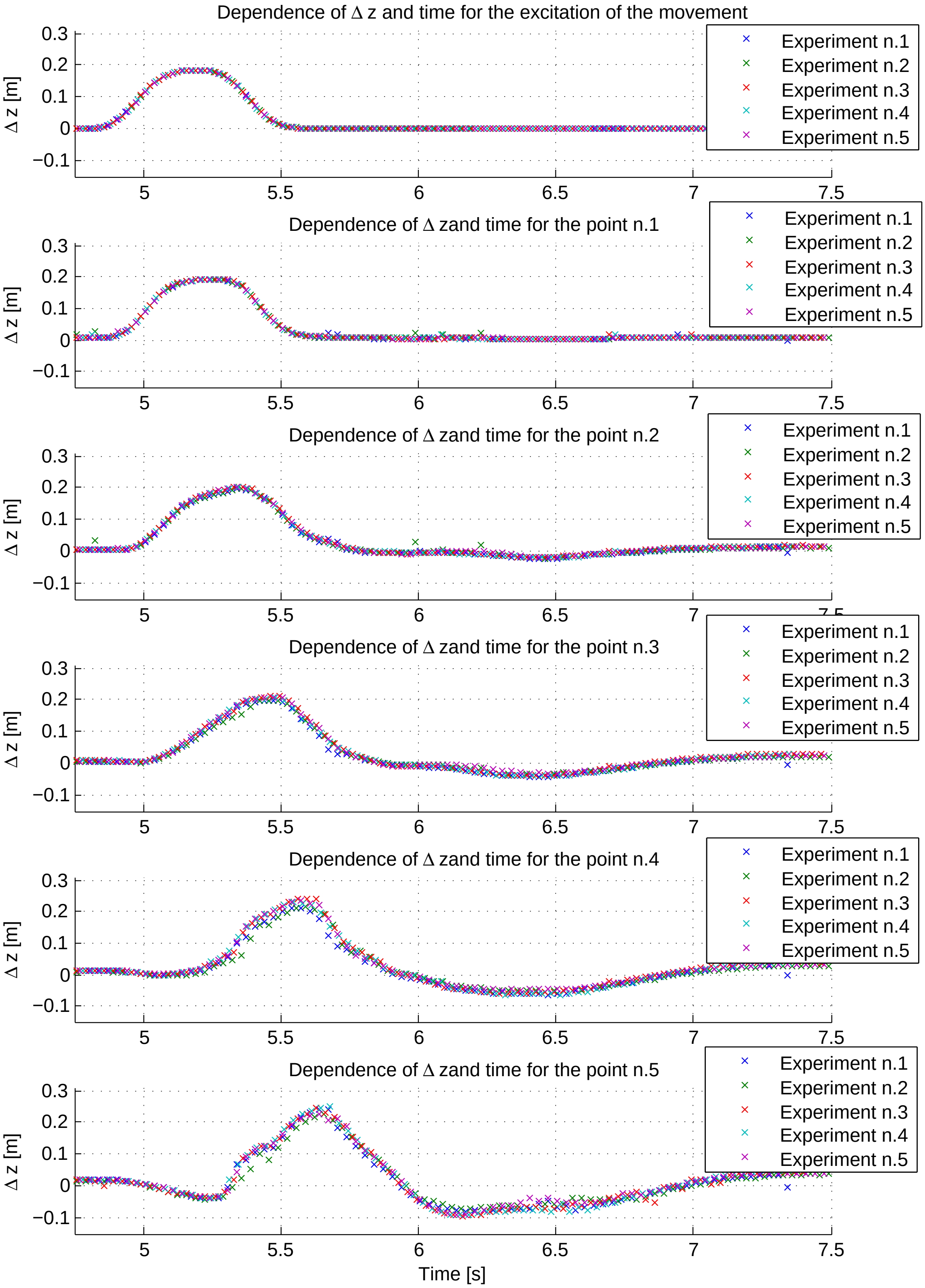


Figure 6.1. Dependence of the position of points and time. Experiment of several measurements of one garment, which is held in the same way.

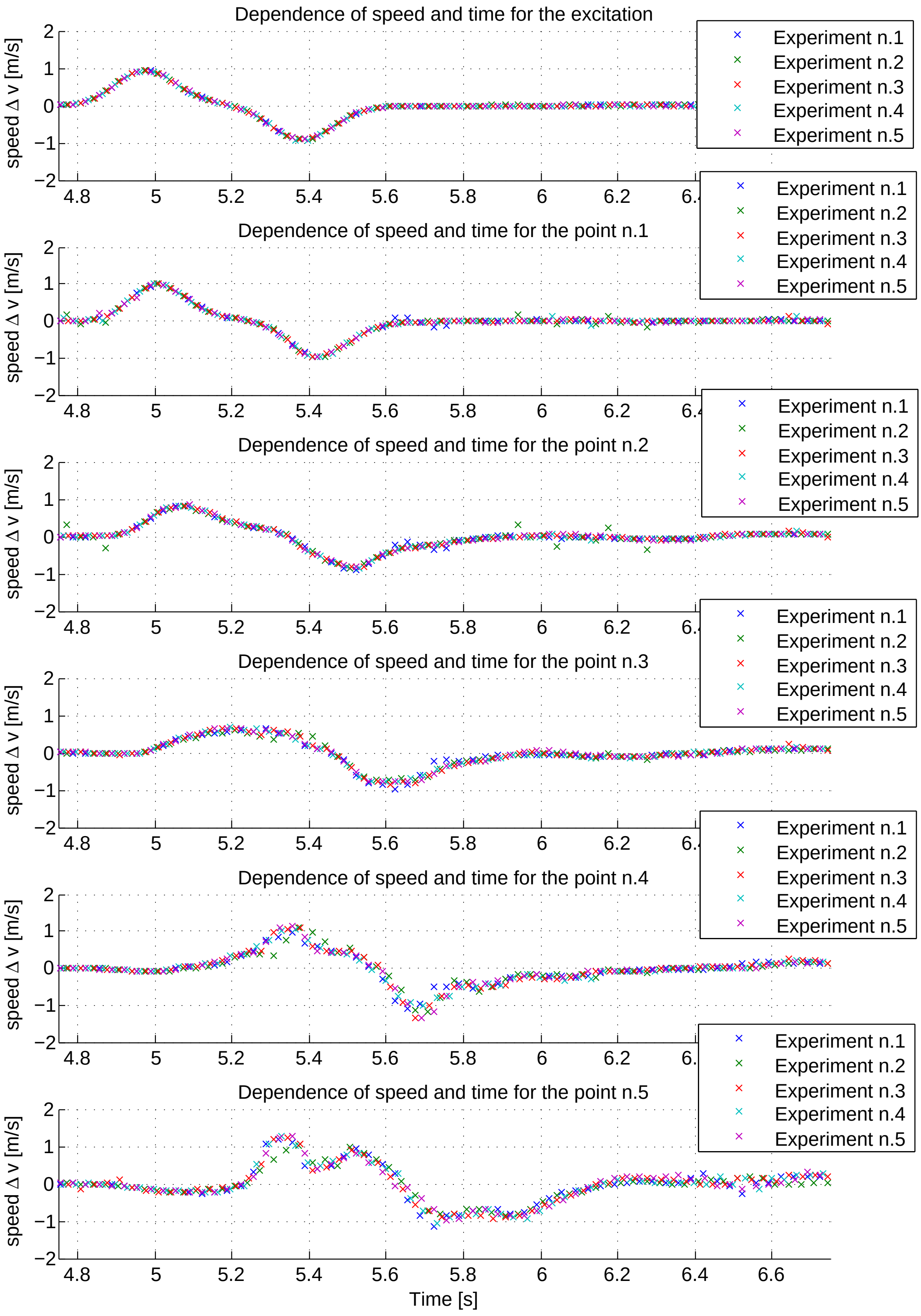


Figure 6.2. Dependence of the speed of points and time. Experiment of several measurements of one garment, which is held in the same way.

Chapter 7
Conclusion

In the introduction of this thesis I have described the methods of obtaining parameters of dynamic model of the garment, based on the mechanical stressing of the garment. These methods are not suitable in our case, so I have developed a method using the robot and the sensors that are available on the robot, which could be used for estimating the parameters of the garment.

The method is based on extraction features from RGB images and depthmaps. The capturing the motion of the garment was performed in two ways. Firstly, RGB video camera captured a silhouette of the garment against the monochromatic background when the garment was moving perpendicular to the camera axis. Secondly, rangefinder captured the garment when garment was moving along the rangefinder axis. I designed an algorithm to compute the position parameters of the garment from the captured data, which could be used for estimating the parameters of the dynamic physical model of the garment.

Due to general use, we separated measuring data, storing data and subsequent processing, so it would be possible to use other methods of processing of the captured data. Data capture and storage is implemented in the robotic operating system ROS. Reading data and extraction of image feautures is implemented in MATLAB.

Within this work, I have captured forty-six experiments with eight different garments. Functionality of an algorithm is documented on selected examples of garments green towel, which was held as shown in the figure 6.1 and 6.2. It turns out that, even when repeating the experiment we were getting very similar data. This behavior corresponds with our expectations so it can be concluded that the procedure is correct. Data was forwarded for further processing (in the thesis [30]). Data are stored on the server of the project CloPeMa, which this work was a part of. The selected data is appended to DVD as described in the chapter 6.

References

[1] CloPeMa clothes perception and manipulation.
`http://clopema.eu/`. Accessed: 2014-03-20.

[2] Kwang-Jin Choi and Hyeong-Seok Ko. Research problems in clothing simulation. *Computer-Aided Design*, 37(6):585–592, 2005.

[3] Jan Bender and Crispin Deul. Adaptive cloth simulation using corotational finite elements. *Computers & Graphics*, 37(7):820–829, 2013.

[4] Yongjoon Lee, Jaehwan Ma, and Sunghee Choi. Automatic pose-independent 3d garment fitting. *Computers & Graphics*, 37(7):911–922, 2013.

[5] Demetri Terzopoulos, John Platt, Alan Barr, and Kurt Fleischer. Elastically deformable models. In *ACM Siggraph Computer Graphics*, volume 21, pages 205–214. ACM, 1987.

[6] David E Breen, Donald H House, and Michael J Wozny. Predicting the drape of woven cloth using interacting particles. In *Proceedings of the 21st annual conference on Computer graphics and interactive techniques*, pages 365–372. ACM, 1994.

[7] Pascal Volino, Martin Courchesne, and Nadia Magnenat Thalmann. Versatile and efficient techniques for simulating cloth and other deformable objects. In *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, pages 137–144. ACM, 1995.

[8] David Baraff and Andrew Witkin. Large steps in cloth simulation. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, pages 43–54. ACM, 1998.

[9] J Eischen and Roberto Bigliani. Continuum versus particle representations. *Cloth modeling and animation*, pages 79–122, 2000.

[10] Kwang-Jin Choi and Hyeong-Seok Ko. Stable but responsive cloth. In *ACM SIGGRAPH 2005 Courses*, page 1. ACM, 2005.

[11] Christiane Luible and Nadia Magnenat-Thalmann. The simulation of cloth using accurate physical parameters. *CGIM 2008, Insbruck, Austria*, 2008.

[12] Sueo Kawabata, R Postle, and Masako Niwa. *Objective specification of fabric quality, mechanical properties and performance.* Textile Machinery Society of Japan, 1982.

[13] George K Stylios. New measurement technologies for textiles and clothing. *International Journal of Clothing Science and Technology*, 17(3-4):3–4, 2005.

[14] Kristina Ancutienė, Eugenija Strazdienė, and Anastasija Nesterova. The relationship between fabrics bending rigidity parameters defined by kes-f and fast equipment. *Materials Science= Medžiagotyra*, 16(4):346–352, 2010.

[15] George Stylios. *Textile objective measurement and automation in garment manufacture.* Ellis Horwood Ltd, 1991.

[16] Pier Giorgio Minazio. Fast–fabric assurance by simple testing. *International Journal of Clothing Science and Technology*, 7(2/3):43–48, 1995.

[17] Kiran S Bhat, Christopher D Twigg, Jessica K Hodgins, Pradeep K Khosla, Zoran Popović, and Steven M Seitz. Estimating cloth simulation parameters from video. In *Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 37–51. Eurographics Association, 2003.

[18] Hatem Charfi, André Gagalowicz, and Rémi Brun. Determination of fabric viscosity parameters using iterative minimization. In *Computer Analysis of Images and Patterns*, pages 789–798. Springer, 2005.

[19] Václav Jahoda.

[20] YASKAWA Electric Corporation.

[21] ASUS Xtion PRO LIVE Specifications.
http://www.asus.com/Multimedia/Xtion_PRO_LIVE/. Accessed: 2014-05-03.

[22] Morgan Quigley, Ken Conley, Brian Gerkey, Josh Faust, Tully Foote, Jeremy Leibs, Rob Wheeler, and Andrew Y Ng. Ros: an open-source robot operating system. In *ICRA workshop on open source software*, volume 3, 2009.

[23] Patrick Mihelich et al. ROS Wiki documentation of rosbag package.
http://wiki.ros.org/rosbag. Accessed: 2014-04-16.

[24] Ben Charrow. A small library for reading ROS bags in Matlab.
https://github.com/bcharrow/matlab_rosbag. Accessed: 2014-02-02.

[25] Milan ŠONKA, Václav HLAVÁČ, and Roger BOYLE. *Image processing, analysis, and machine vision.* Thomson, edition 3 edition, 2008. ISBN: 0-495-08252-X.

[26] Robert M Haralock and Linda G Shapiro. *Computer and robot vision.* Addison-Wesley Longman Publishing Co., Inc., 1991.

[27] Douglas Schwarz.

[28] Liefeng Bo and Kevin Lai. depthtocloud.m - convert depth image into 3d point cloud.
rgbd-dataset.cs.washington.edu/software/depthToCloud.m. Accessed: 2014-02-02.

[29] Tully Foote et al. ROS Wiki documentation of tf package.
http://wiki.ros.org/tf. Accessed: 2014-04-16.

[30] Petr Pecha. Bachelor thesis [in Czech]: Modelování látky v pohybu. Supervisor: Vladimír Smutný, Published: 2014.

Appendix A
Content of Enclosed DVD

dictionary	content
neoramic-bc.pdf	this thesis
/dataset	chosen set of measurements of the garments
/matlab	MATLAB scrips of dataprocessing
/src	Python scrips of manipulation with robot and services

Table A.1. Content of enclosed DVD.

Appendix B
List of the Measured Dataset

In the tables B.2 and B.3 a column `garment` refers to the figure 4.1 and column `holding` refers to the figure C.1. The means of these column could be read from the `column` name of the file as well according to naming of files described in the chapter 4.2.3. The last column `dvd` shows which of these measured data are stored on the enclosed DVD.

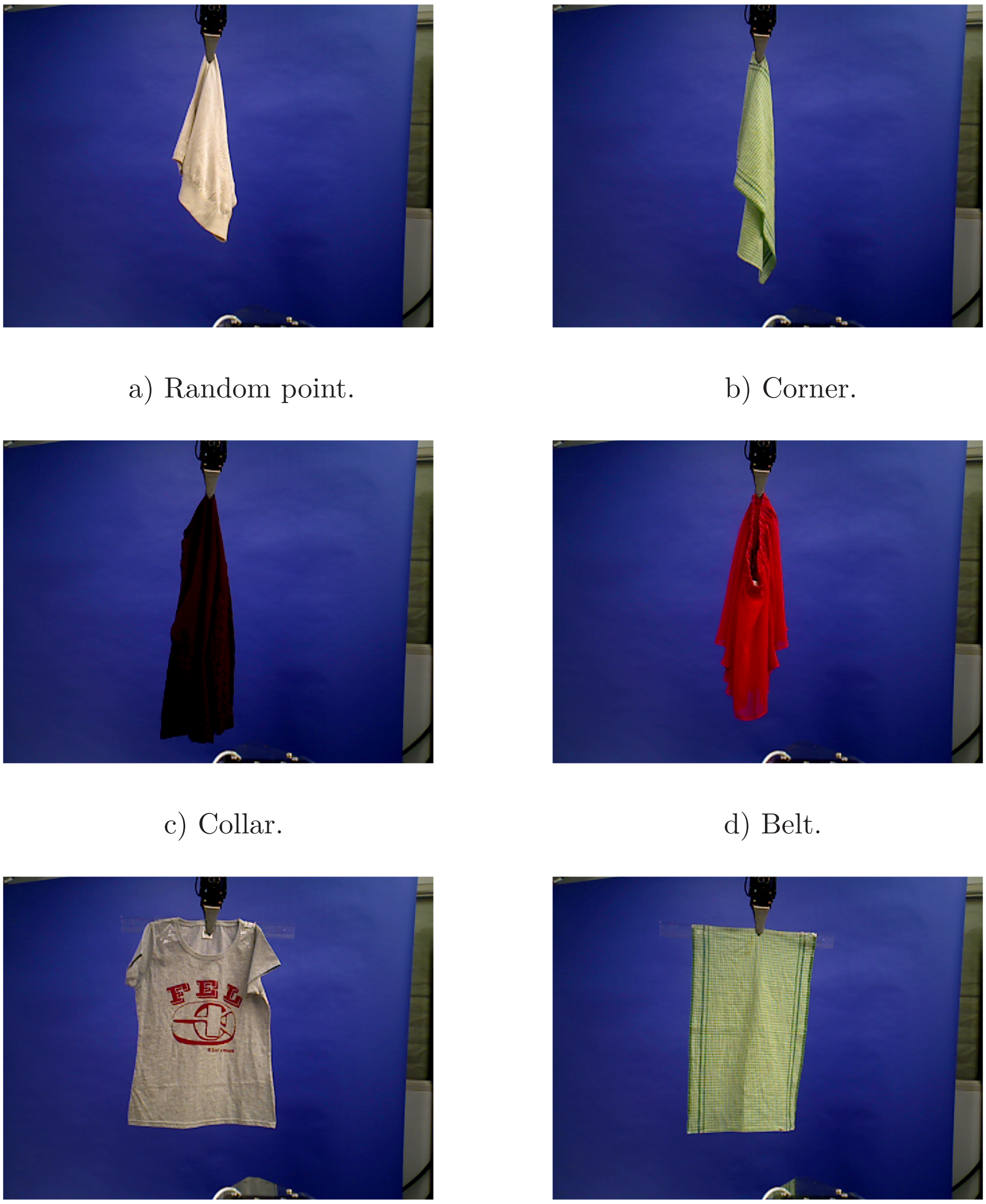
n	name of the file	garment	holding	dvd
1	BlackShorts_Belt_pi16_0.5_R1	e	d	no
2	BlackShorts_Free_pi16_0.5_R1	e	a	no
3	BlueShirt_Collar_pi16_0.5_R1	a	c	no
4	BlueShirt_Free_pi16_0.5_R1	a	a	no
5	FelShirt_Collar_pi16_0.5_R1	b	c	yes
6	FelShirt_Free_pi16_0.5_R1	b	a	yes
7	FelShirt_Ruler_pi16_0.5_R1	b	e	yes
8	GreenTowel_Corner_01_pi16_0.5_R1	d	b	yes
9	GreenTowel_Corner_02_pi16_0.5_R1	d	b	no
10	GreenTowel_Corner_03_pi16_0.5_R1	d	b	no
11	GreenTowel_Corner_04_pi16_0.5_R1	d	b	no
12	GreenTowel_Corner_05_pi16_0.5_R1	d	b	no
13	GreenTowel_Free_pi16_0.5_R1	d	a	no
14	GreenTowel_Ruler_01_pi16_0.5_R1	d	f	yes
15	GreenTowel_Ruler_02_pi16_0.5_R1	d	f	yes
16	GreenTowel_Ruler_03_pi16_0.5_R1	d	f	yes
17	GreenTowel_Ruler_04_pi16_0.5_R1	d	f	yes
18	GreenTowel_Ruler_05_pi16_0.5_R1	d	f	yes
19	RedPullover_Collar_pi16_0.5_R1	g	c	no
20	RedPullover_Free_pi16_0.5_R1	g	a	no
21	RedSkirt_Belt_pi16_0.5_R1	c	d	yes
22	RedSkirt_Free_pi16_0.5_R1	c	a	no
23	WhiteBlackShirt_Collar_pi16_0.5_R1	f	c	no
24	WhiteBlackShirt_Free_pi16_0.5_R1	f	a	no
25	WhitePullover_Collar_pi16_0.5_R1	h	c	no
26	WhitePullover_Free_pi16_0.5_R1	h	a	no

Table B.2. List of the measured dataset of depth maps.

n	name of the file	garment	holding	dvd
1	BlackShorts_Belt_pi16_0.5_B1	e	d	no
2	BlackShorts_Free_pi16_0.5_B1	e	a	no
3	BlueShirt_Collar_pi16_0.5_B1	a	c	no
4	BlueShirt_Free_pi16_0.5_B1	a	a	no
5	FelShirt_Collar_pi16_0.5_B1	b	c	yes
6	FelShirt_Free_pi16_0.5_B1	b	a	yes
7	GreenTowel_Corner_01_pi16_0.5_B1	d	b	yes
8	GreenTowel_Corner_02_pi16_0.5_B1	d	b	yes
9	GreenTowel_Corner_03_pi16_0.5_B1	d	b	no
10	GreenTowel_Corner_04_pi16_0.5_B1	d	b	no
11	GreenTowel_Corner_05_pi16_0.5_B1	d	b	no
12	GreenTowel_Free_pi16_0.5_B1	d	a	no
13	RedPullover_Collar_pi16_0.5_B1	g	c	no
14	RedPullover_Free_pi16_0.5_B1	g	a	no
15	RedSkirt_Belt_pi16_0.5_B1	c	d	yes
16	RedSkirt_Free_pi16_0.5_B1	c	a	no
17	WhiteBlackShirt_Collar_pi16_0.5_B1	f	c	no
18	WhiteBlackShirt_Free_pi16_0.5_B1	f	a	no
19	WhitePullover_Collar_pi16_0.5_B1	h	c	no
20	WhitePullover_Free_pi16_0.5_B1	h	a	no

Table B.3. List of the measured dataset of RGB images.

Appendix C
Cases of Holding of the Garment



a) Random point.

b) Corner.

c) Collar.

d) Belt.

e) With the ruler.

f) Second example with the ruler

Figure C.1. Cases of the holding the garment.

Appendix D
Abbreviations

1D,2D,3D,...	One Dimension, Two Dimensions, Three Dimensions, ...
CTU	Czech Technical University in Prague.
KESF	The Kawabata Evaluation System for Fabric is used to measure the mechanical properties of fabrics.
FAMOUS	Fabric Automatic Measurement and Optimisation Universal System.
FAST	Fabric Assurance by Simple Testing.
fps	Frame Per Second.
MOCAP	MOtion CAPture is the process of recording the motion of objects or people.
OS	Operating System.
P2P	A Peer-To-Peer it’s type of decentralized network.
PLMS	Pucker Laser Measurement System.
RGB	The aditive color model of using Red, Green and Blue colors of lights to create or capture the required color.
ROS	The Robot Operating System - an open source system is used for cotrol robots.
URI	Uniform Resource Identifier.