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

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

Draft: 9. 5. 2014

Acknowledgement / Declaration

Foremost, I would like to thank to Ing. Pavel Krsek, Ph.D. ...

do 11.5.

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

Tento...

do 11.5.

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

Překlad titulu: Získání příznaků z obrazu pohybující se látky

This...

do 11.5.

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

Contents /

Tables / Figures

Chapter 1
Introduction

1.1 Motivations

This bachelor thesis is 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].

1.2 Goals

The whole CloPeMa project is based on the manipulation of clothes (garments). Simplified dynamic physical model of real garment should be useful for these manipulations e.g. for simulating movement or collision detection. Creating virtual model of garment depend on its parameters. This thesis has goal to design method of measurement and extraction of image features for obtaining parameters which will lead to construct the model.

1.3 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 garment [2] (including modern metod of simulating [3–4]). A lot of simulators and 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 from a real parameters of garments (fabrics) obtained e.g. by KESF etc. [9, 11]. But simulation of garment is not a point of this work.

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. 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 and ways (depends on current physical property). The KESF produces curves depending on the applied force. The Kawabata instruments test with high accuracy: compression, pure bending rigidity, roughness, shear, surface friction and tensile [12–14].

Very similiar to the Kawabata's System 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 and recovery behaviour while FAST system determines deformation level at a single point on the deformation curve, so FAST system cannot measure hysteresis [14]. Another differences are that the KEFS use different equipment for each property. The FAST are more properties measure on one equipment, so the number of equipments are reduced [13].

The Fabric Automatic Measurement and Optimisation Universal System (FAMOUS) is faster method of "manual" measurement. A complete suite of measurement take less than five minutes [13].

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 measurement 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 substances, 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. [17] has developed the method 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].

- Zmínit a více rozepsat použití Mocapu [18]

Therefore, we propose which parameters we will need for build a simple dynamic physical model and we propose easiest way to obtain these parameters. We think that for such a simplified model, the parameters are well estimated from a moving garment, for which this model we want to build. This movement will cause the robot and we will capture the movement according to available equipment of robot (chap. 2.1), thus we use the RGB camera and rangefinder.

Chapter 2
Description of Workplace and of the Software

[sec:workplace] 2.1 Workplace

2.1.1 Manipulator

The base is composed of two robotic manipulator arm Motoman MA1400. First arm is called as `r1` (or also appears like `R1`). Second arm is similarly marked `r2` (`R2`). The arms `r1` and `r2` are placed on the turntable. The turntable is rotated about an axis known as `external axis` (or `Ext.` or possibly as axis `13`). Location of arms and rotating around the `Ext.` axis can be better seen from (figure 2.2).

[fig:cturobot]

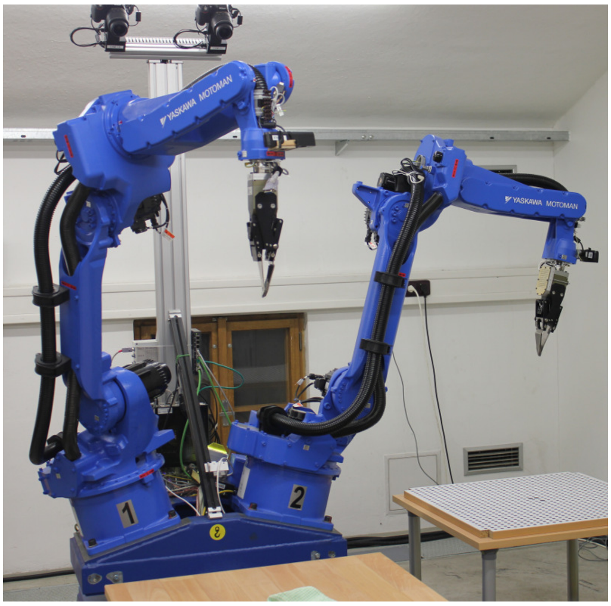
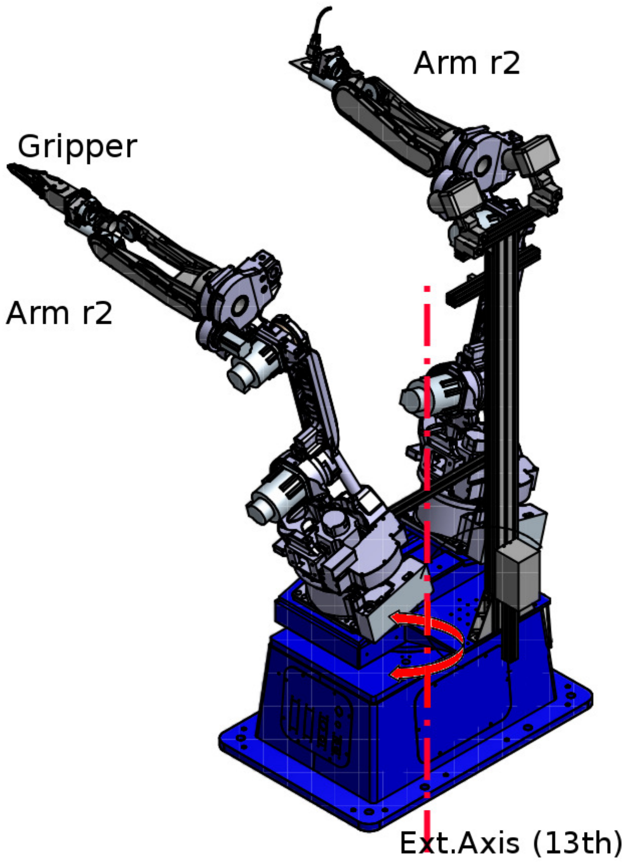


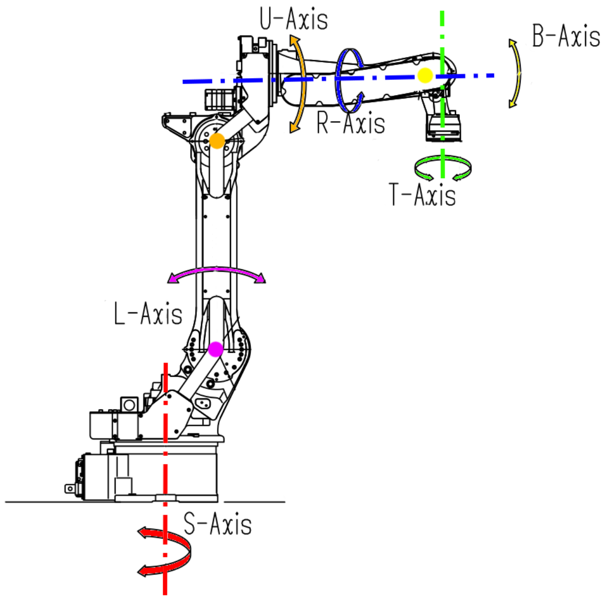
Figure 2.1. Manipulator of CloPeMa project location at CTU

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 description of single the 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`).



[fig:motomanAndTable]

Figure 2.2. Identification of arms and location of external axis.



[fig:motomanAxis]

Figure 2.3. Description of robotic arm Motoman MA1400 - axis.

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 University of Genoa. The gripper is composed of two supple fingers. Supple 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]. **použít SOFT místo SUPPLE?**

[fig:gripper]

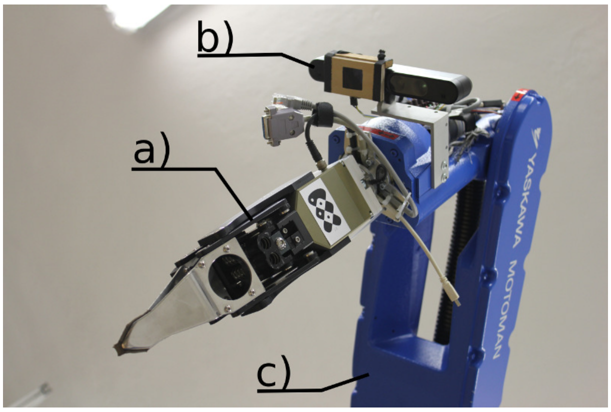


Figure 2.4. End effector (gripper). a) gripper, b) sensor Asus Xtion, c) end of arm on which the gripper is mounted.

[secc:camera]

2.1.3 Sensors

The robot has been setup with a variety of sensors:

- Robot Binocular-Vision Head
  - The robot head comprises two Nikon D5100 DSLR cameras. These are mounted on two pan and tilt units. The head provides the robot system with high resolution 3D points clouds [1].
- Photometric Stereo Gripper-Mounted Sensor
  - The is a small scale close range stereo-pair 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 robot has two to three these sensors — one on the each arm [1].

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 [19]. In CloPeMa project is used Ubuntu (Debian-based Linux OS) as a host operating system.

[secc:rosintro]

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 [19].

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 [19].

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 basucally a set of tools for recording from and playing back to ROS topics [20]. With help of this tool we can record all or just some chosen topics. The chosen topics are passed like 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 too.

Chapter 3
Way of getting data

[sec:requirements]

3.1 Goals of experiment

The requirement on the experiment 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. Movement of hanging garment will be causes the movement of the manipulator gripper that holds garment. Based on the sensors that we have available, we have chosen:

- simplest movement, which could give us the necessary data. Data that could be used to obtain the features of the dynamic model of garment. This movement is the movement of garment in the plane, ideally excited by moving gripper of a garment in a straight line (line segment),
- two types of motion capturing:

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

3.2 Realisation

Both arms of manipulator are used for the experiment. One arm causes a movement of garment that is held by the gripper of this arm. Xtion sensor (2.1.3), located on the second arm, is used for capturing the movement.

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

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

We decided capture the movement of garment in two ways. We use both possibilities of Xtion sensor — capturing RGB images and depthmaps too. When garment is moving perpendicular to the optical axis, will be captured RGB images. When garment is moving along the optical axis, will be captured depthmaps. We need background subtraction for processing RGB images. Background subtraction needs RGB image without garment on foreground.

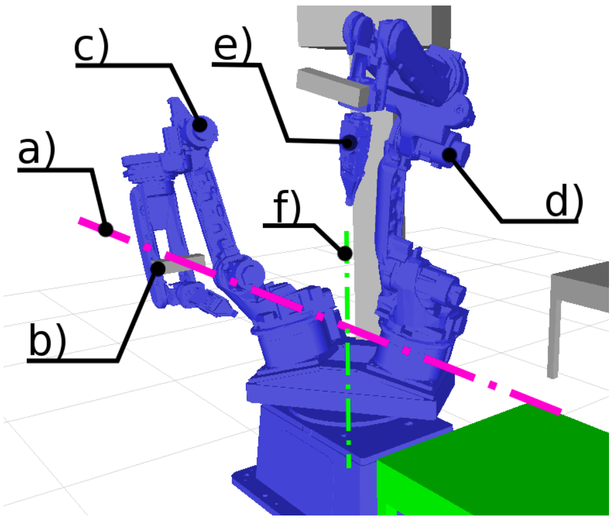
Another limitation is the spatial limitation, such that it is not possible to place the camera `xtion` in the appropriate position to capture RGB images (ie, the position where the gripper with garment moves perpendicular to the optical axis) and then the camera `xtion` move to position suitable for 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 movements fulfilled the conditions for sensing with each sensors (perpendicular position vs. along the optical axis). **OBRÁZEK - Ale jaký?**

[sec:caprgb]

3.3 Capturing of RGB

[secc:extrgb]

3.3.1 External Axis



[fig:OptOsa]

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.

`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 experiment.

[secc:r1rgb]

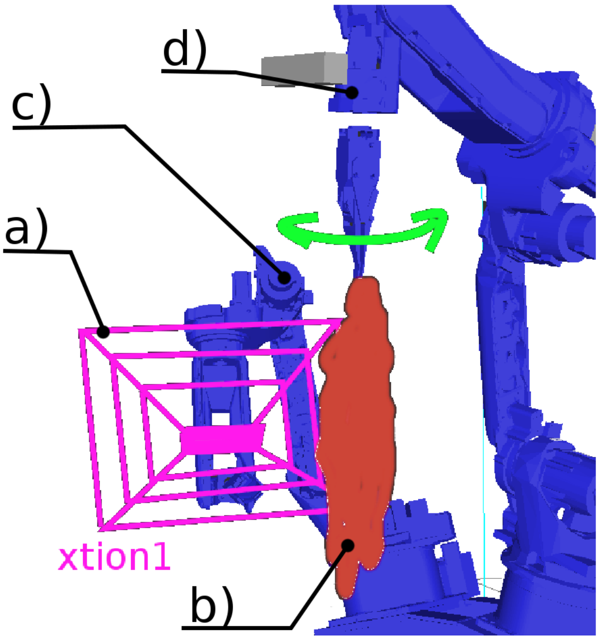
3.3.2 Arm with Xtion Sensor

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 experiment.

[subsec:refRGB]

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 movement of garment. The arm `r2` is in a position which it can perform movement required for the experiment (chap. 3.1). The arm `r2` makes movement so as garment moved perpendicularly to the optical axis. The arm `r2` makes a desired movement 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 movement is movement mooted in the figure 3.2.



[fig:kolmoOptOsy]

Figure 3.2. Suggestion of movements 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.

[sec:capDepth]

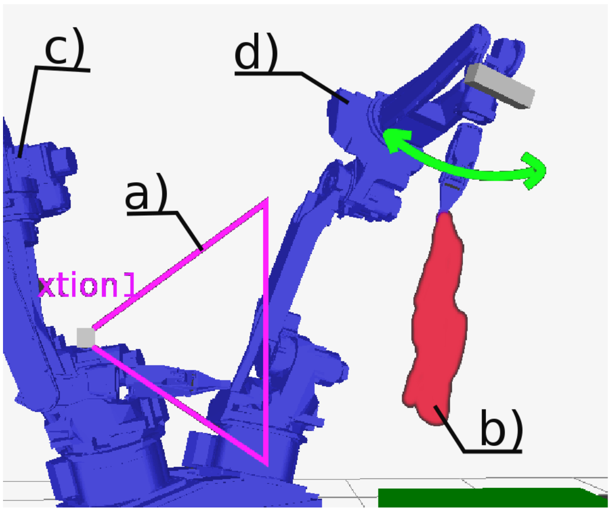
3.4 Capturing of Depth maps

3.4.1 External axis

`Ext.` axis (axis 13) is rotated as in the case of capturing of RGB (chap. 3.3.1).

3.4.2 Arm with Xtion sensor

The arm `r1` is into same position as in case of capturing of RGB (chap. 3.3.2).



[fig:rovnoOptOsy]

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

3.4.3 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 movement of garment. The arm `r2` is in a position which it can perform movement required for the experiment (chap. 3.1). The arm `r2` makes movement so as garment moved along to the optical axis. The arm `r2` makes a desired movement with the garment so that it rotates about an axis `R` certain angle and will return back to initial position. For better describe of the movement is movement mooted in the figure 3.3.

sec:captureBackground]

3.5 Capturing Background Image

This position is used for captire a reference image of background, for improve results of the experiment. The captured image is used for filtering background (background substraction) from RGB image. More to background substraction is deal in chapter ??.

3.5.1 External axis

`Ext.` axis (axis `13`) is rotated as in the case of capturing of RGB (chap. 3.3.1).

3.5.2 Arm with Xtion sensor

The arm `r1` is into same position as in case of capturing of RGB (chap. 3.3.2).

3.5.3 Arm with Garment

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.

Chapter 4
Data Structure

For the purpose of the experiment is good data processed offline. It is therefore important to store the measured data to data structure and then in MATLAB calculate the parameters that are important for the experiment (chap. 5). If the results of experiment are good and quick, will be the calculation in the future transformed from MATLAB to the ROS.

4.1 Format of Recorded Data

Data is stored by using rosbag tool (chap. 2.2.1) in the format `.bag` to the folder set in the `local_options.py` file [1]).

[sec:topics]
4.2 Topics

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 experiment. 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
```

4.3 Measured Data Set of the Garments

4.3.1 Structure of Data Set

Zde bude popsána datová sada a kde bude uložena

4.3.2 Format of Names of Recorded Files

Recorded files are stored under different names accord to the form `name_speed_AX.bag` (table 4.1).

4.3.3 Description of the Garments

Zde bude výčet některých použitých látek jako hmotnosti, rozměry ...

[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

name	choosen file name by user
speed	choosen speed of manipulator
A	axis, which was executed movement **R** or **B** (figure 2.3)
[explanation] X	number of topics file

Table 4.1. Explanation of format file name.

[chap:dataproc]

Chapter 5
Data Processing

[sec:loadToMatlab]

5.1 Load Data to the MATLAB

Data are processed offline in MATLAB. The offline procesing is only for research purposes. If it turns out that the proposed method of extraction of features from moving garment and methods of model building a dynamic physical model of the garment[1]) are good, methods will be implemented directly into the ROS and processed online. Data format `bagfile` are read using the `matlab_rosbag` tool (from [21]). 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 [2]) as are read chosen topics for recording data (chap. 4.2). After loading the data into MATLAB data are grouped into cells by topic.

```
SCRIPT loader
    Initalize;
    Load BAGFILE;
    Load BACKGROUNDBAGFILE for background substraction;
    Load TOPICS from topics.txt;
    For each TOPICS from FILE make matrix of cell;
END.
```

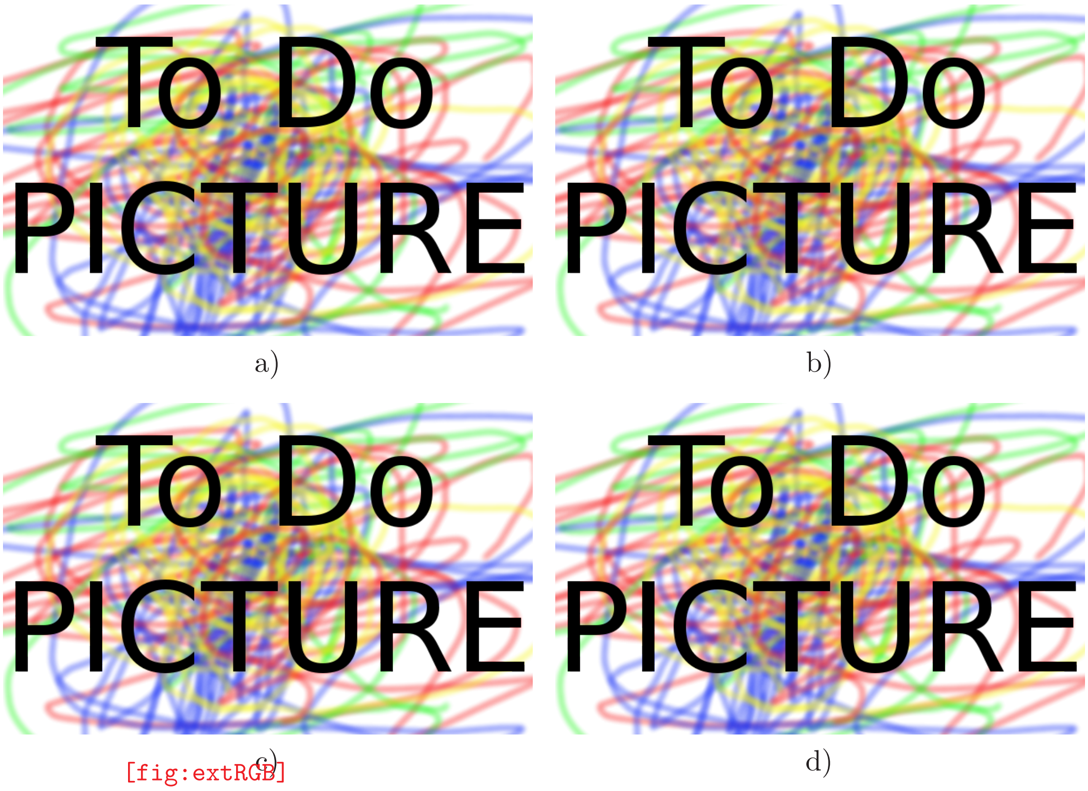
Next steps in the case of RGB images and Depth maps are different.

[sec:rgb]

5.2 Extraction of Features from RGB

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` like single row vector. After loading data into MATLAB (chap. 5.1) data are converted from row vector to matrix $\mathbf{I}_{R,G,B}(u, v)$ of RGB images with dimensions 480×640 px ($u \times v$). Raw RGB data shows figure 5.1.

[1]) Create a dynamic physical model of the garment is not the point of this work.
[2]) `path_to_workspace/clopema_cvut/clopema_collect_model_data/matlab/topics/topics.txt`



[fig:extRGB]

Figure 5.1. Illustration RAW RGB images.

5.2.1 Rectification of RGB

- Popsat, jak opravím RGB snímek + obrázek

[sec:bagSubRGB]

5.2.2 Background Subtraction

In the chapter 3.5 is described a way of capturing background image. Respectively was not captured a single image, but was captured a sequence of images. The sequence of images $\mathbf{B}^i_{R,G,B}(u,v)$ is averaged to $\overline{\mathbf{B}}_{R,G,B}(u,v)$ (5.1). Together with mean value of RGB background is computed corrected sample standard deviation $\sigma_{R,G,B}(u,v)$ (5.2).

[eq:averageRGB]

$$\overline{\mathbf{B}}_{R,G,B}(u,v) = \sum_{i=1}^{N} \frac{\mathbf{B}^i_{R,G,B}(u,v)}{N} \tag{5.1}$$

[eg:deviationRGB]

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

From sequence RGB images $\mathbf{I}^i_{R,G,B}(u,v)$ is computed a sequence of silhouettes $\mathbf{S}^i(u,v)$ (5.3).

[eq:backgroundSubtraction]

$$\mathbf{S}^i(u,v) = \begin{cases} 1, & \text{if} \begin{cases} & \left|\mathbf{I}^i_R(u,v) - \overline{\mathbf{B}}_R(u,v)\right| > \sigma_R(u,v); \\ \text{or} & \left|\mathbf{I}^i_G(u,v) - \overline{\mathbf{B}}_G(u,v)\right| > \sigma_G(u,v); \\ \text{or} & \left|\mathbf{I}^i_B(u,v) - \overline{\mathbf{B}}_B(u,v)\right| > \sigma_B(u,v); \end{cases} \\ 0, & \text{otherwise.} \end{cases} \tag{5.3}$$

$$\mathbf{S}^i(u,v) = \begin{cases} & \left|\mathbf{I}_R^i(u,v) - \overline{\mathbf{B}}_R(u,v)\right| > \sigma_R(u,v); \\ \text{or} & \left|\mathbf{I}_G^i(u,v) - \overline{\mathbf{B}}_G(u,v)\right| > \sigma_G(u,v); \\ \text{or} & \left|\mathbf{I}_B^i(u,v) - \overline{\mathbf{B}}_B(u,v)\right| > \sigma_B(u,v); \end{cases}$$

- Způsob filtrace proti pozadí - napsat vzorec
- Popsat i použité morphologické operace pro zkvalitnění siluety (možná vlastní secc)
- Obrázek siluety 4 pcs

5.2.3 Finding End of Gripper

- Popsat, jak naleznu oblast, kterou opisuje chapadlo při hýbání s látkou a jak z tohoto pohybu naleznu konec chapadlo v obraze
- Obrázek s vyznačenou kružnicí a bodem jako koncem gripperu

5.2.4 Finding Central Curve of Garment

- Napsat, jak hledám osu bramboroidu
- Popsat zde zavrhnuté metody

  - Kostra grafu + obrázky + proč jsem to nepoužil
  - Střed dle y osy + obrázek + proč jsem to nepoužil

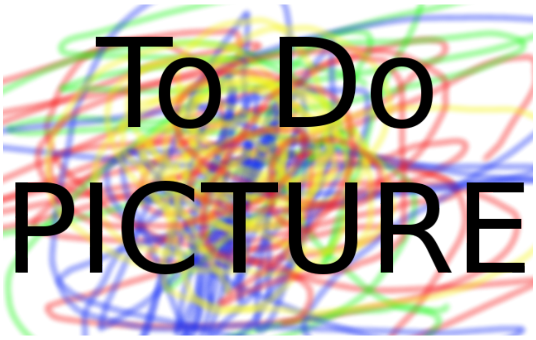
- Obrázky postupného nalezení osy bramborouidu (při aproximaci udělat více obrázků)

5.2.5 Finding Mathematical Features from RGB

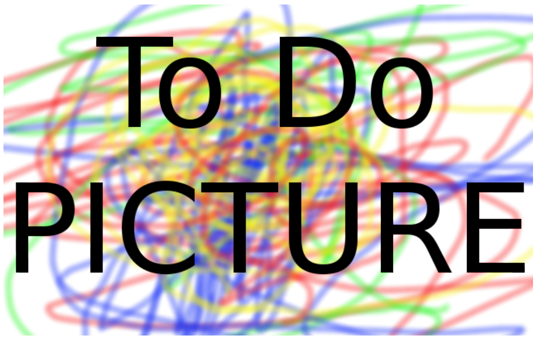
- Popsat, jak z osy bramboroidu naleznu body, které předávám jako výstup
- Obrázky se siluetou a v ní s body

5.3 Extraction of Features from Depth Map

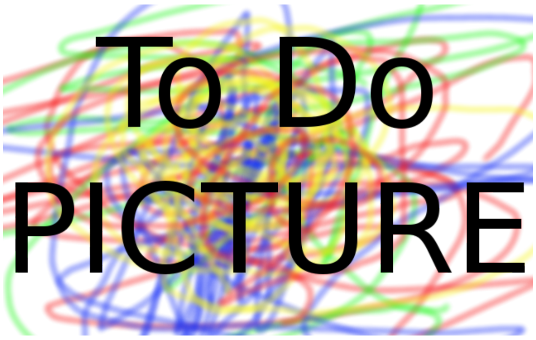
The sequence of images of depth maps is captured with the xtion camera (chap. 2.1.3). Depth maps are in a sufficient resolution 640×480 px (width×height). Depth maps are stored in the `bagfiles` like single row vector. After loading the data into MATLAB (chap. 5.1) data are converted from row vector to matrix $\mathbf{C}(u, v) = z$ of depth map with dimensions 480×640 px ($u \times v$). The depth maps data shows figure 5.2.



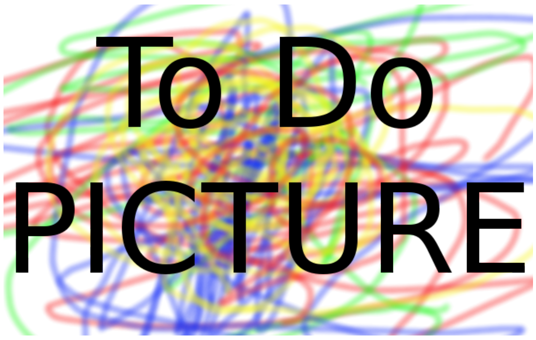
a)



b)



[fig:extDepth] c)



d)

Figure 5.2. Illustration RAW Depth Map from Rangefinder.

5.3.1 Convert Depth Map to 3D points

Together with data depth map is also stored calibration parameters. From these calibration parameters comes calibration matrix $\mathbf{K}$, where f_x and f_y are focal lengths and (c_x, c_y) is a principal point (matrix form eq. (5.4)).

[eq:calib]

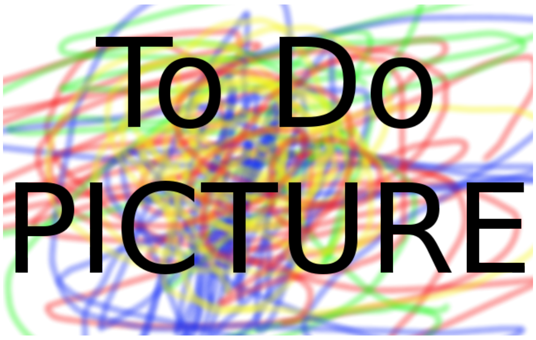
$$\mathbf{K} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \tag{5.4}$$

Using the calibration matrix $\mathbf{K}$ can be depth map $\mathbf{C}(u, v)$ converted into 3D points matrix $\mathbf{M}_{x,y,z}(u, v)$. The matrix $\mathbf{M}_{x,y,z}(u, v)$ is made by equation (5.5). Algoritmus used for these convertion was inpirated by [22].

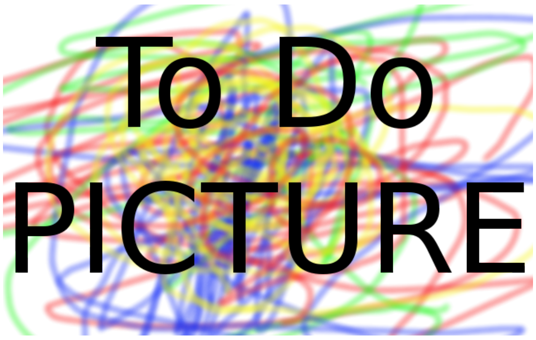
[eq:get3Dpoints]

$$\begin{aligned} \mathbf{M}_z(u, v) &= \frac{\mathbf{C}(u, v)}{1000} \\ \mathbf{M}_x(u, v) &= \frac{\mathbf{M}_z(u, v) \cdot (u - c_x)}{f_x} \\ \mathbf{M}_y(u, v) &= \frac{\mathbf{M}_z(u, v) \cdot (v - c_y)}{f_y} \end{aligned} \tag{5.5}$$

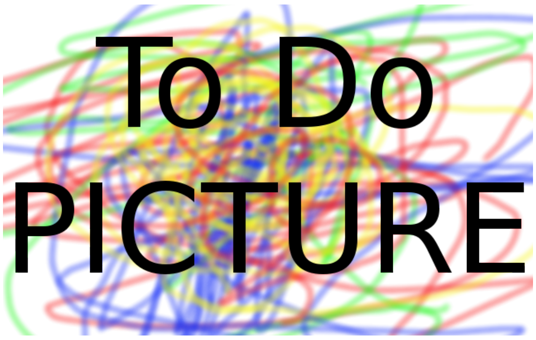
The data from 3D points are shown in the figure 5.3.



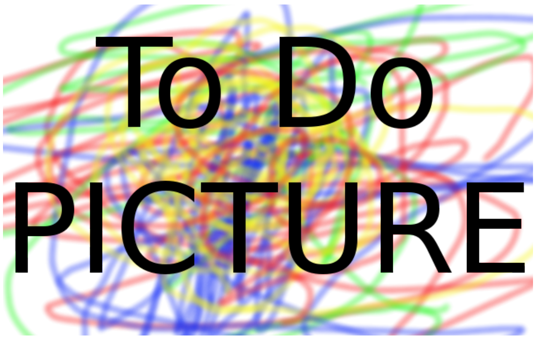
a)



b)



c)



d)

[fig:pointsDepth]

Figure 5.3. Illustration of 3D Points.

5.3.2 Filtering by Depth of Area

The figure 5.3 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. From the measurements, we know that the gripper of the arm `r2` with garment is located **napsat přesnou hodnotu** m before the xtion camera on the arm `r1` at its default measurement position. After several experiments, it was found that the minimum and maximum of z-axis value on the garment have the following values: $min_{garment}(z) = 0.8$ m and $max_{garment}(z) = 1.16$ m, thus we know that the garment moves between these values. Maximum deviation Δz from the z-axis value of endpoint $r2_z$ is computed by equation (5.6).

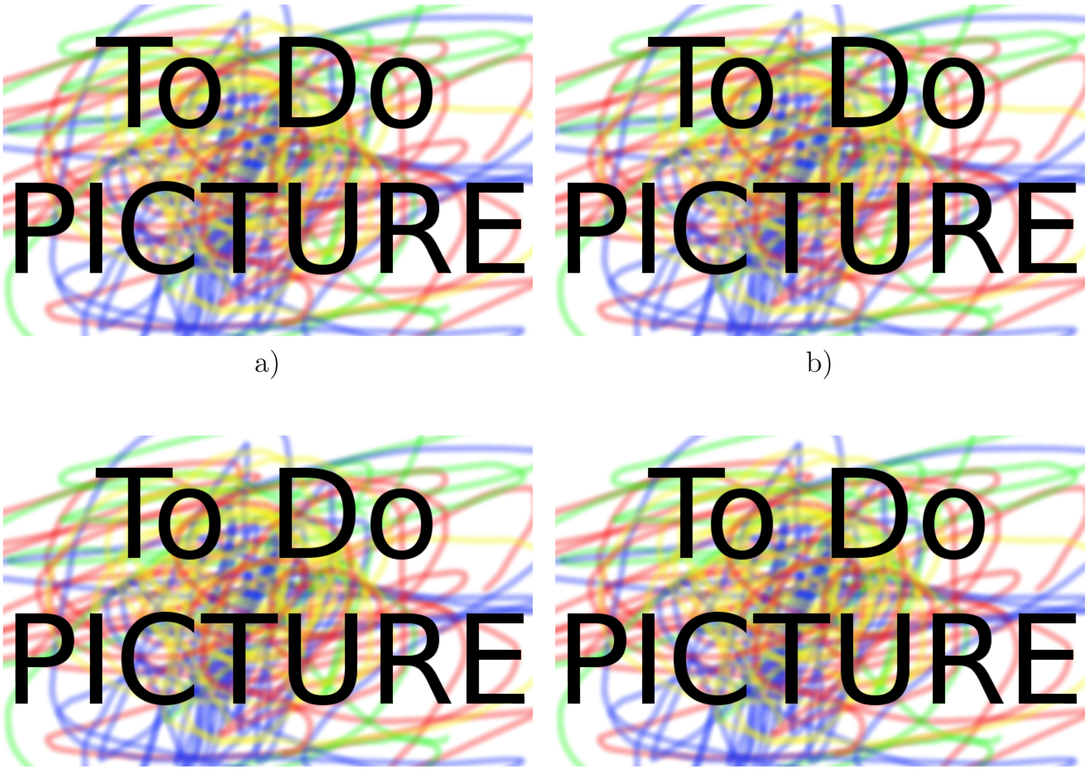
[eq:minmax]

$$\Delta z = max(abs(r2(z) - min_{garment}(z)), abs(r2(z) - max_{garment}(z))) \tag{5.6}$$

If the deviation is known, matrix $\mathbf{M}_{x,y,z}(u,v)$ (from (5.5)) can be filtered using the matrix equation (5.7). The filted data are shown in the figure 5.4. You can see the differences between figures 5.2 and 5.4. Moreover, this method of "background substraction" is much robust than method of background substraction of RGB images ??.

[eq:filter3D]

$$\mathbf{M}_{x,y,z}(u,v) = \begin{cases} \mathbf{M}_{x,y,z}(u,v), & \text{if } r2_z - 3 \cdot \Delta z \leq \mathbf{M}_z(u,v) \leq r2_z + 3 \cdot \Delta z; \\ \text{NaN}, & \text{otherwise.} \end{cases} \tag{5.7}$$



a) b) c) d)

[fig:pointsFiltering]

Figure 5.4. Illustration of 3D points after Filtering by Distance.

[secc:findEE]

5.3.3 Finding End of the Gripper

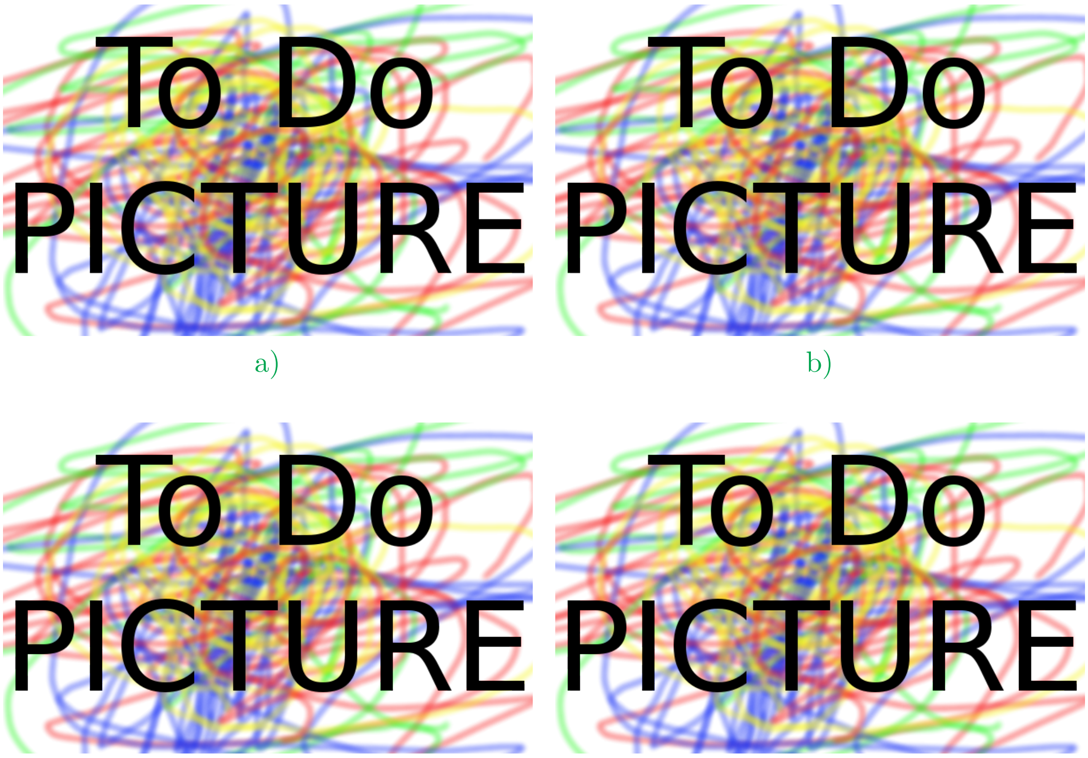
In the figure 5.6 are not shown only points on the garment. The figure shows points of the gripper of arm `r2` 2.1.3 too. As has been said in chapter 3.4 the movement of the garment is caused by this gripper. Finding the coordinates of the end point of the gripper is important not only for distinguishing the gripper from the garment, but also for detecting excitation movement of garment.

- NUTNO PROKONZULTOVAT!
- Nalezení gripperu
- Probrat na konzultaci, jelikož se k tomuto nalezení váže vytvoření listeneru a publisheru pro topic

[eq:endEffector]

$$ee_{x,y,z} \tag{5.8}$$

Founded gripper is shown in the figure 5.5.



a) b)

c) d)

[fig:find3DGripper]

Figure 5.5. Illustration of 3D Points with Founded Gripper.

5.3.4 Finding Points

- Napsat proč hledám tyto body !!!!!!!!
- NUTNO PROKONZULTOVAT!

First of all is reduced the 3D points matrix. Points over the gripper endpoint $ee_{x,y,z}$ (from chap. 5.3.3) are not needed anymore. Then is set a width Δx of garment which is used for next computing. **napsat důvody proč.** Matrix of points $\mathbf{M}_{x,y,z}(u,v)$ are reduced to $\tilde{\mathbf{M}}_{x,y,z}(u,v)$ (5.9).

$$\tilde{\mathbf{M}}_{x,y,z}(u,v) = \begin{cases} \mathbf{M}_{x,y,z}(u,v), & \text{if } (\mathbf{M}_y(u,v) < ee_y)\&(ee_x - \frac{\Delta x}{2} < \mathbf{M}_x(u,v) < ee_x + \frac{\Delta x}{2}); \\ \text{NaN}, & \text{otherwise.} \end{cases} \tag{5.9}$$

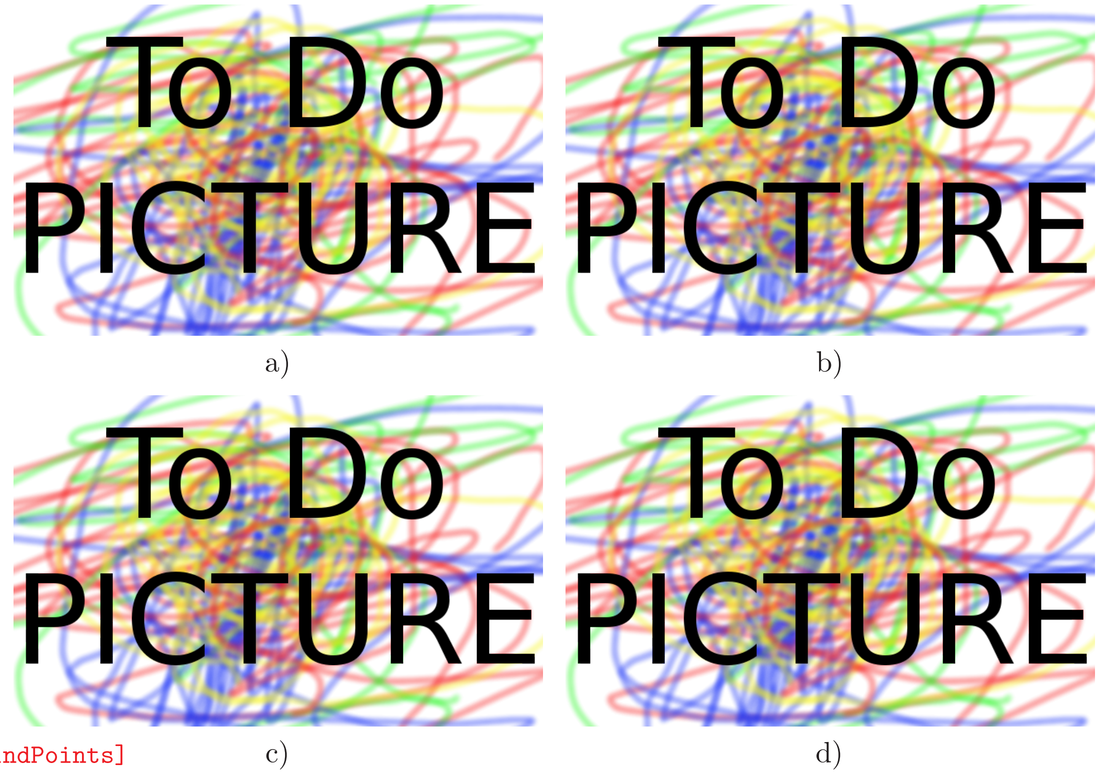
[eq:reducedMatrixDepth]

The reduced points of garment are resampled and averaged to the sample points $\mathbf{S}_{x,y,z}(m)$ (5.10).

[eq:sampleDepth]

$$\mathbf{S}_{x,y,z}(m) \tag{5.10}$$

The length of garment is sumarized according to sample points and divided by the number of required points. Search points are found at a constant distance given length of garment. Founded points are shown in the figure 5.6.

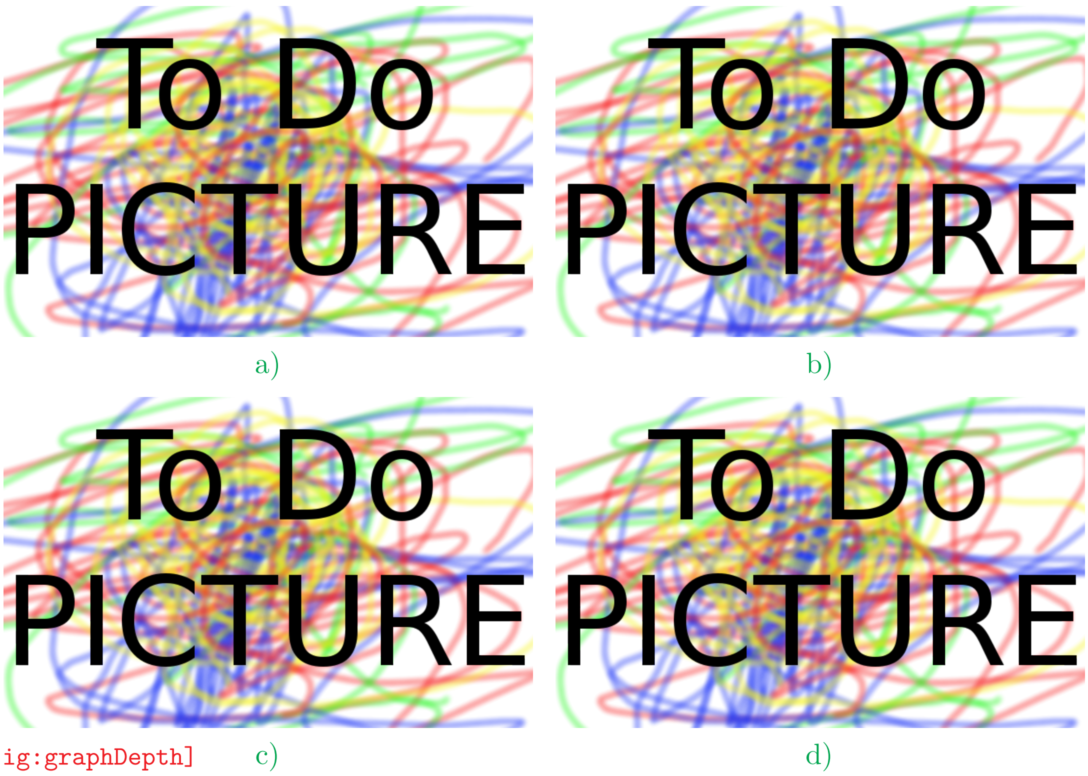


[fig:findPoints]

Figure 5.6. Illustration of 3D points with Points Found on the Garment.

5.3.5 Finding Mathematical Features from Depth Map

- Nalezení a vyplivnutí bodů ke zpracování



[fig:graphDepth]

Figure 5.7. Dependence of the speed of points on the time.

Chapter 6
Results

- Provést měření typu: 5x jsem naměřil po sobě stejným způsobem stejnou látku (provést u více látek) a srovnat, zdali jsou si výsledky podobné.
- Naměřit více různých látek a zjistit, zda-li jsou si látky podobné (neměli by být).
- Provést pro RGB i DepthMap data
- Vyrobit a vložit grafy

Chapter 7
Discussion

- Diskutovat použitelnost
- Napsat, co by se dalo zlepšit, případně i jak
- Podařilo se mi:
- Vyšlo mi:
- Funguje to tak a tak:
- Je to tak a tak rychlé:
- Je to tak a tak přesné:
- Tady jsou typické případy, kde to zafungovalo:
- Tady jsou typické případy, kdy to selhalo:
- Nejvíce si cením:
- Uvést nápady, které jsem nestihl realizovat jako možné pokračování
- Případné vynechání této kapitoly a diskutování v závěru

do 11.5.

Chapter 8
Conclusion

- V práci je: ...
- Hlavní úspěchy jsou: ...
- Důležitými výsledky jsou: ...
- Podařilo se: ...
- Za nejdůležitější výsledek považuji:
- Možnost vynechání kapitoly DISCUSSION a uvedení jejího obsahu sem
- Pohled do budoucna (přeformulovat, změnit, rozšířit):
  - V případě, že se ukáže tento způsob sběru dat a tvorba modelu (odkaz na jinou bc.práci) užitečnou, bylo by dobré naprogramovat celý tento postup i s tvorbou modelu v operačním systému ROS, aby nebylo třeba dalších výpočetních nástrojů (MATLAB).
- Rekapitulovat naplnění všech bodů práce

do 11.5.

References

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

hoi2005research] [2] Kwang-Jin Choi and Hyeong-Seok Ko. Research problems in clothing simulation. *Computer-Aided Design*, 37(6):585–592, 2005.

der2013adaptive] [3] Jan Bender and Crispin Deul. Adaptive cloth simulation using corotational finite elements. *Computers & Graphics*, 37(7):820–829, 2013.

ee2013automatic] [4] Yongjoon Lee, Jaehwan Ma, and Sunghee Choi. Automatic pose-independent 3d garment fitting. *Computers & Graphics*, 37(7):911–922, 2013.

1987elastically] [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.

n1994predicting] [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.

no1995versatile] [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.

baraff1998large] [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.

en2000continuum] [9] J Eischen and Roberto Bigliani. Continuum versus particle representations. *Cloth modeling and animation*, pages 79–122, 2000.

[choi2005stable] [10] Kwang-Jin Choi and Hyeong-Seok Ko. Stable but responsive cloth. In *ACM SIGGRAPH 2005 Courses*, page 1. ACM, 2005.

e2008simulation] [11] Christiane Luible and Nadia Magnenat-Thalmann. The simulation of cloth using accurate physical parameters. *CGIM 2008, Insbruck, Austria*, 2008.

ta1982objective] [12] Sueo Kawabata, R Postle, and Masako Niwa. *Objective specification of fabric quality, mechanical properties and performance.* Textile Machinery Society of Japan, 1982.

[stylios2005new] [13] George K Stylios. New measurement technologies for textiles and clothing. *International Journal of Clothing Science and Technology*, 17(3-4):3–4, 2005.

010relationship] [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.

lios1991textile] [15] George Stylios. *Textile objective measurement and automation in garment manufacture*. Ellis Horwood Ltd, 1991.

[minazio1995fast] [16] Pier Giorgio Minazio. Fast–fabric assurance by simple testing. *International Journal of Clothing Science and Technology*, 7(2/3):43–48, 1995.

t2003estimating] [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.

05determination] [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.

[quigley2009ros] [19] 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.

[rosbag] [20] Patrick Mihelich et al. ROS Wiki documentation of rosbag package. `http://wiki.ros.org/rosbag`. Accessed: 2014-04-16.

[rosbagtool] [21] Ben Charrow. A small library for reading ROS bags in Matlab. `https://github.com/bcharrow/matlab_rosbag`. Accessed: 2014-02-02.

atlabpointcloud] [22] 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.

Appendix A
Specification

Appendix B
Content of included DVD

- Databaze látek
- Získané výsledky z nich
- Natočená videa
  - Videa z processingu - různé stupně a různé látky
  - Video z pracoviště - sběr dat
- Tato práce
- Všechny scripty
- (Manuál na sběr dat CZ)
- ReadMe

do 11.5.

Appendix C
Other Images

- U všech obrázku níže udělat alternativu (lépe dvě) a tu vložit do přílohy a odkázat na ní

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.
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.

Appendix E
Brief Manual to Get Data Manually

- Přeložit návod z CZ do EN

do 11.5.