AALBORG UNIVERSITY
STUDENT REPORT

Vision, Graphics and Interactive Systems
Master Thesis - 2020

Vision-Based Ergonomic Adaptation of Robotic Movements in Human-Robot Collaboration

Marike Koch van den Broek

This report is compiled in LaTeX. Additionally is Python, yEd, and Inkscape used to code, draw figures, and charts.

Vision, Graphics and Interactive Systems
Aalborg University
http://www.aau.dk

AALBORG UNIVERSITY
STUDENT REPORT

Title:
Vision-Based Ergonomic Adaptation of Robotic Movements in Human-Robot Collaboration

Theme:
Interactive Systems

Project Period:
01.02.2019 - 02.07.2020

Project Group:
Group No. 1049

Participants:
Marike Koch van den Broek

Supervisor:
Thomas B. Moeslund

Number of Pages: 117

Number of Appendices: 8

Date of Completion:
2 July 2020

Abstract:

Although the general welfare has improved the working environment for shop-floor workers throughout a variety of industries, the conditions are not necessarily living up to the standards of the general welfare. Some shop-floor workers are still exposed to physically straining jobs, which through challenges in design or business strategies has not caught up with societal tendencies. An example of such a job is the processing and handling of meat by butchers in the meat packing industry, where the incidence of Musculoskeletal Disorders (MSDs) is high. In an attempt to investigate how a combination of robots and vision technology can improve the working conditions for the butcher, this project explores the possibility of vision-based ergonomic adaptation of the robotic movements in Human-Robot Collaboration (HRC). The relevant methods and technologies are investigated and an adaptive system is implemented. Furthermore, a method for adapting robotic movements according to two Ergonomic Risk Factors (ERFs), posture and repetitiveness, is proposed.

Preface

The project report was written by Marike Koch van den Broek as her Master Thesis during completion of the Master of Vision, Graphics and Interactive Systems. The project work was conducted in the time period from February 2019 to July 2020 and corresponds to the workload of 50 ECTS.

Reading Directions
This part contains directions for readers regarding the way references, figures, bibliography and appendices are structured, and how they should be understood.

References

- The reference referring to the single sentence is set in the end of the sentence before the full stop.
- The reference referring to a paragraph is put after the full stop in the last sentence in the paragraph.

Figures and Tables

- Most of the figures and tables have been made by the author. Such figures do not have a source.
- If a figure or a table is not the author's ownership, permission to include the element in the thesis has been requested and granted by the rightful owner and the source is given in a footnote.
- Lists of figures and tables are provided in the end of the report before the bibliography.

Bibliography

- The bibliography can be found in the end of the report.
- All entries in the bibliography are written in alphabetical order.
- The bibliography entries are referenced in the text with an author-year key. If no information about year is available, it is referenced by author only.
- Each bibliography entry is structured in a following way: author, year, title, journal, volume, pages, type, URL, etc. The amount of given information might vary depending on available information.

Appendices

- All appendices are located in the end of the report.
- Each appendix is enumerated by a letter.

Written by:

Marike Koch van den Broek

Contents

Glossary

AAU Aalborg University.
ACMP Augmented Cellular Meat Production.
DMRI Danish Meat Research Institute.
DOF Degrees of Freedom.
EMG Electromyography.
FOV Field of View.
GUI Graphical User Interface.
HRC Human-Robot Collaboration.
HRI Human-Robot Interaction.
IMU Inertial Measurement Unit.
IS Initial System.
MSDs Musculoskeletal Disorders.
NERPA Novel Ergonomic Postural Assessment Method.
REBA Rapid Entire Body Assessment.
ROS Robot Operating System.
RULA Rapid Upper Limb Assessment.

Chapter 1

Introduction

This project focuses on exploring computer vision for ergonomic assessment with the purpose of monitoring and adapting the movements of an assistive collaborative robot in order to improve the working environment for human workers. The specific applicational context for this project is the working environment of butchers working in a mass production meat packing facility. This chapter will examine the technical challenges related to the problem as well as give a brief overview of the motivational background to the problem.

1.1 Motivation

Musculoskeletal Disorders (MSDs) can be defined as any health problems ranging from light to irreversible in the locomotor apparatus. MSDs include conditions that affect muscles, ligaments, tendons, nerves, and blood vessels. Conditions induced or aggravated by work are considered Work related MSDs (WMSDs). (Chander and Cavatorta, 2017; Punnett and Wegman, 2004) In 2017 MSDs was the most frequently reported occupational illness in Denmark (see Figure 1.1) (Arbejdstilsynet, 2018). Many shop floor workers in industry are subject to strain and, ultimately, risk factors for MSDs. This also applies to the workers in the food industry in Denmark, especially the butchers working in facilities of the meat packing industry. For several decades the industry has been associated with a high rate of injuries and illnesses such as MSDs.(Grant and Habes, 1997; Moore and Garg, 1998; Tappin et al., 2008) In fact, a study conducted on behalf of the Danish trade union for workers in the food industry, "Fødevareforbundet NNF," shows that butchers were responsible for just below half of the work related injuries reported on average yearly among their members in the years 2010-2013. Just about two thirds of these reports were occupational illnesses. Likewise, a report by the Danish Working Environment Authority reveals that the meat packing industry had the highest incidence of occupational illnesses in Denmark in 2017. Furthermore, the meat packing industry had the highest incidence of MSDs reported among any of the investigated industries in Denmark in 2017; an amount 5 times higher than the average amount reported per industry MSDs (see Figure 1.2). (Arbejdstilsynet, 2018)

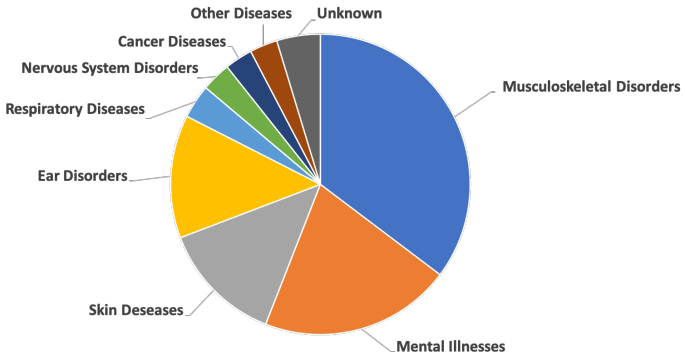


Figure 1.1: The distribution of the reported occupational illnesses in 2017 in Denmark.[1]

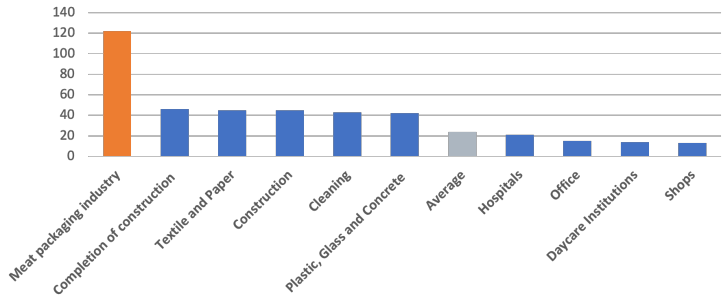


Figure 1.2: Number of reported MSDs per 10.000 employed (incidence) in 2017 in Denmark distributed across industry groups.[2]

Several sources report that this specific group of workers is at high risk for MSDs such as Carpal Tunnel Syndrome, Rotator cuff syndrome and Epicondylitis(Moore and Garg, 1998; Gorsche et al., 1999; Bjarne Thomsen (Chef consultant), Tommy Jensen (Federal consultant), Per Hansen (Federal secretary), 2019). Moreover, according to Fødevareforbundet NNF, every second member has consumed pain medication or had the need for pain treatment within a time period of three moths as a consequence of their work (Toxverd, 2018). Though not stated, with the workers in the meat packing industry having the largest incidence of work related injuries and illnesses, it is likely that they also are the group of members consuming the most pain medication. There are several factors that cause the work to be straining, and which makes the possibilities of improvements to the current situation an intricate affair. Firstly, different parts of the work carried out requires a lot of force exertion by the worker e.g. handling and cutting the meat. Secondly, the work conducted by the butchers in the slaughterhouse is highly repetitive due to the need of a high production rate. The workers stand in lines along the side of conveyer belts and work at a high pace for hours with limited possibility for taking small stretch breaks in between. The workers are paid based on a basic amount combined with piecework payment (Fødevareforbundet NNF and DI, 2017). This means that the butchers earn more the harder they work, which gives them motivation to maintain a high work pace ignoring the need to lower the pace or take breaks.

The competition on the international meat market puts pressure on the production costs and, thereby, sets a need for a high production rate. In the mean time, the danish welfare and the trade union push for good conditions and environments for the workers, which urges the employers to limit their demands and compromise to such an extend that a conflict of interests

[2]Figure was created based on data from Arbejdstilsynet (2018).

arises.

The management at the slaughterhouses and the trade union are aware of these challenges and do have measures in place for limiting the risk factors affecting the workers. During introduction and education of a new worker, specific instructions on the right course of action and the right movements are given. Furthermore, ergonomic inspections of the employed workers are conducted, where it is also assessed whether the workers follow the instructions or should be given elaborate lessons. Finally, in order to minimise the repetitiveness of the work, rotations have been implemented such that the workers are not doing the exact same movements for an entire shift but have some variation. (Bjarne Thomsen (Chef consultant), Tommy Jensen (Federal consultant), Per Hansen (Federal secretary), 2019)

Because of these well known factors and conflict of interests, investigations and experiments have been conducted thought the last decades in order to examine bodily activity, forces and strain during the working positions. Some of the investigations have merely provided insights or mapped the conditions under which the butchers work while other have lead to information which can be used for the suggestion of specific improvements to the work environment and tools. (Grant and Habes, 1997; Pontonnier et al., 2014; Moore and Garg, 1998)

Changes which will increase the pressure on the business strategies of the management of the slaughter houses and influence the static power balance between cost and conditions are expected to play a roll in the foreseeable future. As a consequence, investigations are made into how the meat production industry of the future should look like including new technologies, more flexible and autonomous solutions, and improvements to the working environments. An example of a project concerned with such investigations is the Augmented Cellular Meat Production (ACMP) project.

1.2 The ACMP Project

The Augmented Cellular Meat Production (ACMP) project is hosted and lead by the Technological Institute branch Danish Meat Research Institute (DMRI) in collaboration with several partners: the workers union "Fødevareforbundet NNF," educational and research institutes, such as Aalborg University (AAU) and DTU, and management of select slaughterhouses among other[3]. The project is financed by 'Innovation Fund Denmark.' The purpose of the project is to innovate and reform the traditional meat packing industry. This is done by investigating possibilities for implementing technology in order to make the facilities more flexiblility, improve the quality, minimise mistakes and implement solutions that support the butchers in their work. As a part of the latter improving the working environment and lessen the load for the butchers is considered. The project includes three work packages each concerned with implementing assistive systems based on new technologies and higher level of automation in the meat packing industry.

The first package is called 'Quality inspection and contaminant detection' and focuses on detecting flaws and contaminants with x-rays. Furthermore, it is concerned with finding the optimal visual way to communicate this information to the butchers who are responsible for the removal of the contaminants.

"Belly trimming with handling assistance" is the title of the second work package. This work package is concerned with obtaining the perfect thickness of the fat layer on the belly pieces and providing support for the workers during the handling of the pieces. This includes tasks such as automatic unloading of belly pieces, presentation of pieces to the butchers, providing

[3]Information about the ACMP project is provided by The Danish Meat Research Institute

visualisation of the optimal trim with updated feedback during trimming, and recognition of the belly pieces. Furthermore this work package includes investigations into how a robot can be utilised for providing the mentioned support.

The third work package, "Augmented carcass processing Assistance," is focussed on cells for meat processing that are flexible and highly automated but keeps the human in the loop. The work package includes investigation into automated processing like parting pork, cutting ears etc. using robots but keeping the human in the loop. One of the aims is to find the best distribution of the work between a human worker and the robot. Other focus points of this work package are investigation of how to handle and adapt to the biological variation in the carcasses, and the human-robot communication.

This master thesis mainly relates to the second work package and to the handling and presentation of the belly pieces. It builds on additional thoughts expressed during meetings between the parties involved with the ACMP project. The thesis focusses on how to improve ergonomic conditions and limit the butchers exposure to ergonomic risk factors during their handling and processing of belly pieces. More specifically, it investigates whether and how an assisting collaborative robot through its movements automatically can facilitate this and adapt to the human operator when it presents the belly pieces to the butcher. In section 1.3 a possible design outline of such a system will be provided.

1.3 System Outline

This section gives an outline of how a possible system that aligns with the thoughts and wants expressed by the working group in the ACMP project could be structured.

A system able to adapt according to human poses is dependent on several components. A basic outline can be seen in Figure 1.3. A system with the abilities described in section 1.2 have to receive some input from the butcher/operator in order to be able to adapt. This could be input actively provided by the operator, but since the aim is to relieve the operator and make the robot responsible for autonomously adapting, monitoring of the human is preferable. This requires sensors of one or more types. Also, a control system probably consisting of several submodules is needed for processing the input received from the sensors, decision making and generation of control output for a collaborative robot. Finally, a robot or manipulator is needed for providing assistance and perform the actual adaptation to the operator.

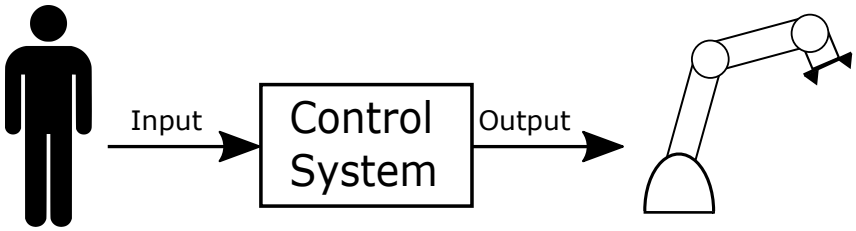


Figure 1.3: The basic system outline.

When taking a closer look at the functional requirements for the control system it is clear that it will consist of several modules each responsible for some specific processes or operations needed in the system. Because the goal is optimisation in relation to ergonomic conditions and as posture is an important aspect of ergonomics, detection of the operator's posture is inevitably important. This sets specific requirements to the monitoring system including both the sensors and the processing module. A posture processing module should be able to calculate information about the operator's posture based on the sensor input. Subsequently, an ergonomic estimation

module should be able to estimate some measure of the ergonomic state of the operator. Based on the ergonomic information a decision making module should find the right course of action in order to minimise strain and ergonomic risk factors based on a robot response model. Lastly, a robot control module should send control information to the robot such that the wanted course of action is carried out. An overview of the control system with the different modules can be seen in Figure 1.4.

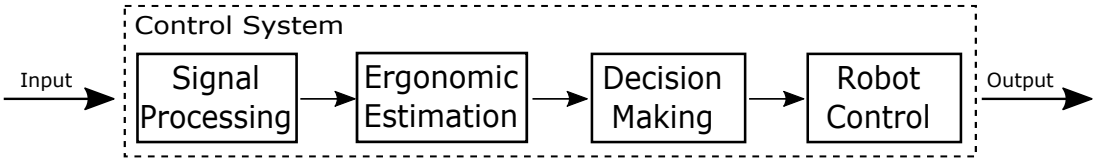


Figure 1.4: The expanded structure of the control system with the different functional modules.

In section 1.4 an investigation of the state of the art methodology in relation to these modules is presented. As the signal processing done in order to obtain a pose estimation is closely linked to the sensors used, these two topics will be addressed together in section 1.4.1. Similarly, the connection between the ergonomic estimations and adaptive behaviour of a robot will be explored in section 1.4.3.

1.4 State of the Art

This section explores the state of the art in regards to different subareas related to the problem and system introduced in section 1.2 and section 1.3.

1.4.1 Human Pose Estimation

This section gives a brief overview of the field of human pose tracking/estimation and some of the general methods within the field.

Through the years, a lot of work has been conducted investigating methods for pose estimation or motion capture of the human body. The motivation behind comes from a wide range of applications such as biomechanical analysis (Dugan and Bhat, 2005), animation for movies and games (Lee et al., 2002), gesture recognition and pose estimation for interaction purposes (Ubisoft; T. C. Tan and Arai, 2011).

The methods that have been implemented varies in respect to the technology used and the performance achieved. The choice of method is dependent on requirements and constrains of the application at hand. While all methods are dependent on some kind of sensing, some are based on camera technology while others are based on e.g. Inertial Measurement Unit (IMU)s. Examples of the latter is Smartsuit Pro from the company Rokoko[4], or the products from the company Xsens[5].

Methods that are vision based also vary. Some methods utilise markers. Examples hereof are the systems offered by Vicon which utilise reflective markers, an infrared strobe and several cameras (Vicon, 2019). Another example is the PhaseSpace Impulse X2 system which uses active LED markers and records using infrared stereocameras (Wang et al., 2015). The markers are matched between the images from the different angles in order to obtain precise positions of the markers in the three dimensional cartesian space. Another example of a marker-based method is presented by Marcon et al. (2017). They use ArUco fiducial markers attached to clothing and observed with a camera in order to obtain the posture of the upper body.

[4]https://www.rokoko.com/en/products/smartsuit-pro
[5]https://www.xsens.com/products/

The Microsoft Kinect cameras with the related software, e.g. Microsoft Kinect SDK[6] or OpenNI tracker[7], are examples of a marker-less vision based method for human pose estimation. Here computer vision algorithms are used to identify key points (Wang et al., 2015). Abobakr et al. (2017) presented a different method using the Kinect camera where joint angles were directly estimated from depth images. Another source combined two Kinect cameras in order to improve the performance of tracking (Yeung et al., 2013).

For a more elaborate overview of human body motion capture methods, please see (Xia et al., 2017; Filippeschi et al., 2017).

1.4.2 Ergonomic Assessment

As the world has developed a larger focus on occupational health and safety, the need for tools for e.g. ergonomic assessment has become more pressing. International competition and changing demographics also result in the idea that sustainable Human Resource Management is needed in the future. Ergonomic considerations and tools are included in the design of products as well as production process and workstation layouts in production. (Schaub et al., 2012; Högberg et al., 2017)

The purpose of ergonomic assessments is to estimate the severity of exposure to ergonomic risk factors typically in order to estimate whether subjects are susceptible to MSDs (McAtamney and Corlett, 1993). Typically, the assessment is conducted including estimations of one or more key aspects that are considered risk factors for MSDs. The aspects encountered in literature are posture, force/load, frequency, repetitiveness, duration, vibration, recovery time, movement velocity etc.(Occhipinti, 1998). The aspect most commonly included in assessments are the first two mentioned; however, the influence of other aspects such as duration and repetition on the risk of MSDs should not be underestimated. (David, 2005; Kroemer, 1989; Descatha et al., 2009; Grandjean and Hünting, 1977)

A variety of ergonomic assessment tools and methods for different purposes have been introduced and improved through the years, and new methods are still being developed (Chander and Cavatorta, 2017). Generally the methods are categorised into the groups: self-reporting, direct measurement and observational methods based on the technique used for measuring (Abobakr et al., 2017; Yazdanirad et al., 2018; Vignais et al., 2013).

Self-reporting techniques, which include questionnaires and interviews, are characterised by being dependent on the subject of the assessment to provide the valuable insights. This means that the method relies on rather subjective experiences and impressions.

The direct measurement techniques rely on precise measurements of the body, positions of the body segments and joints. The need for high precision usually means that one of the methods described in section 1.4.1 where the attachment of units or markers to the body is required for measuring is utilised. The measurements are typically used in combination with biomechanic models of the human body for complex, demanding calculations and simulations. The output of such systems is typically kinetic and dynamic information calculated from masses and kinematic information. As a consequence of the complexity of the models and the calculations, it is currently not possible to process data realtime using these tools. An example of such a tool is Anybody.[8] Systems as these provide an opportunity to get detailed information about the strain in the different joints in the form of force and torque values; however, as a rule it does not offer any analytic parameters in regard to ergonomics such as scores. This means that either an

[6] https://www.microsoft.com/en-us/download/details.aspx?id=44561
[7] https://github.com/ros-drivers/openni_tracker
[8] https://www.anybodytech.com

expert has to assess the ergonomic conditions based on the values, or an additional programme has to be used which is able to calculate scores or indices equivalent to the ones obtained in the observational methods (mentioned below) but on an more informed basis.

In the observational methods non-invasive monitoring of the subjects is done, and tools or metrics are used to evaluate the ergonomics of the observed. Traditionally, a human observer would utilise tools in the form of worksheets either by evaluating immediately at the scene or by evaluating postures at a later time using video recordings. The tools or worksheets utilised in the observational methods are useful as they aid alignment and conformity of assessments conducted by different ergonomists or observers. A study conducted by Eliasson et al. (2017) showed great variance in assessments conducted without tools or structured methods by ergonomists. Even the variance of the same observer assessing the same task with a minimal of three weeks between was unacceptable.

Well known tools include Rapid Upper Limb Assessment (RULA) (McAtamney and Corlett, 1993), Rapid Entire Body Assessment (REBA) (Hignett and McAtamney, 2000), Novel Ergonomic Postural Assessment Method (NERPA)(Sanchez-Lite et al., 2013), NIOSH (National Institute for Occupational Safety and Health) Lifting Equation (Waters et al., 1993, 1994) and the Snook tables (Snook and Ciriello, 1991). While tools such as RULA and REBA are focussed on the general posture assessment under any circumstances, other tools such as NIOSH Lifting Equation, Wisha Lifting Guide and The Snook Tables are focused on analysis in regards to a specific operation, in this case lifting or pushing, pulling and carrying, respectively. More extensive lists have been comprised by David (2005) and Roman-Liu (2014).

In both RULA and REBA the assessments are based on angles observed in the body. Based on defined intervals a score is assigned to the different angles. A final score is found by combining sub-scores found through table lookups, and combining the obtained scores with scores for other aspects such as coupling, load and activity level. See Figure 1.5a.

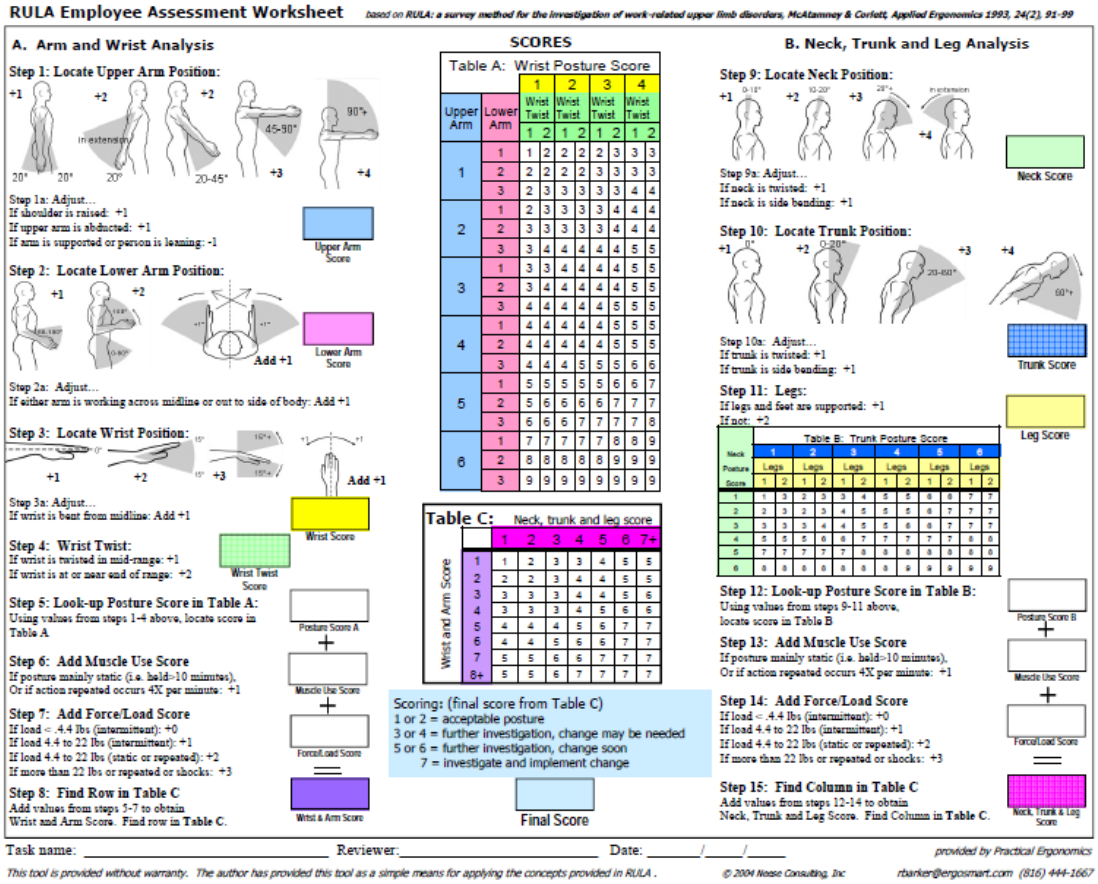
NIOSH Lifting Equation is different in the way that the identified variables are used in equations in order to obtain a risk index and a lifting index as well as a recommended limit for the weight to be lifted in the given task.

The OCRA, an index of exposure to repetitive movement of the upper limbs, is inspired by the NIOSH Lifting Equation in the way that the index is calculated based on a series of other variables. It includes aspects of force, posture, frequency etc. (Occhipinti, 1998).

NERPA was designed to be an improved assessment tool taking its starting point in RULA but modified according to a specific set of ISO standards. This is done in an attempt to improve on some of the RULA method's shortcomings; one of them being the presence of false negatives. (Sanchez-Lite et al., 2013), See Figure 1.5b

In literature there are also examples of tools that are specifically designed for assessment in a constrained context. One of such tools is the Agricultural Lower-Limb Assessment (ALLA) suggested by Kong et al. (2010), which, as the name indicates, was developed with the purpose of assessing the ergonomics of lower-limb postures that occur within agriculture. The method was validated by comparing its compliance with expert assessments to the compliance of other assessment methods applied to the same cases. The study showed that ALLA more accurately matched the expert assessment when evaluating postures in Korean farming than other methods tested; however, the other methods tested were RULA, REBA etc. which are not specialised in lower-limb assessment like ALLA is (Kong et al., 2018). Another example is the ergonomic walkthrough checklist proposed by Bhattacharya et al. (1997) with the object of identifying ergonomic risk factors related to tasks carried out in carpentry. The checklist tool was designed

[9]Source: Neese Consulting, https://www.physio-pedia.com/images/e/e6/RULA.png
[10]Source: Sanchez-Lite et al. (2013)



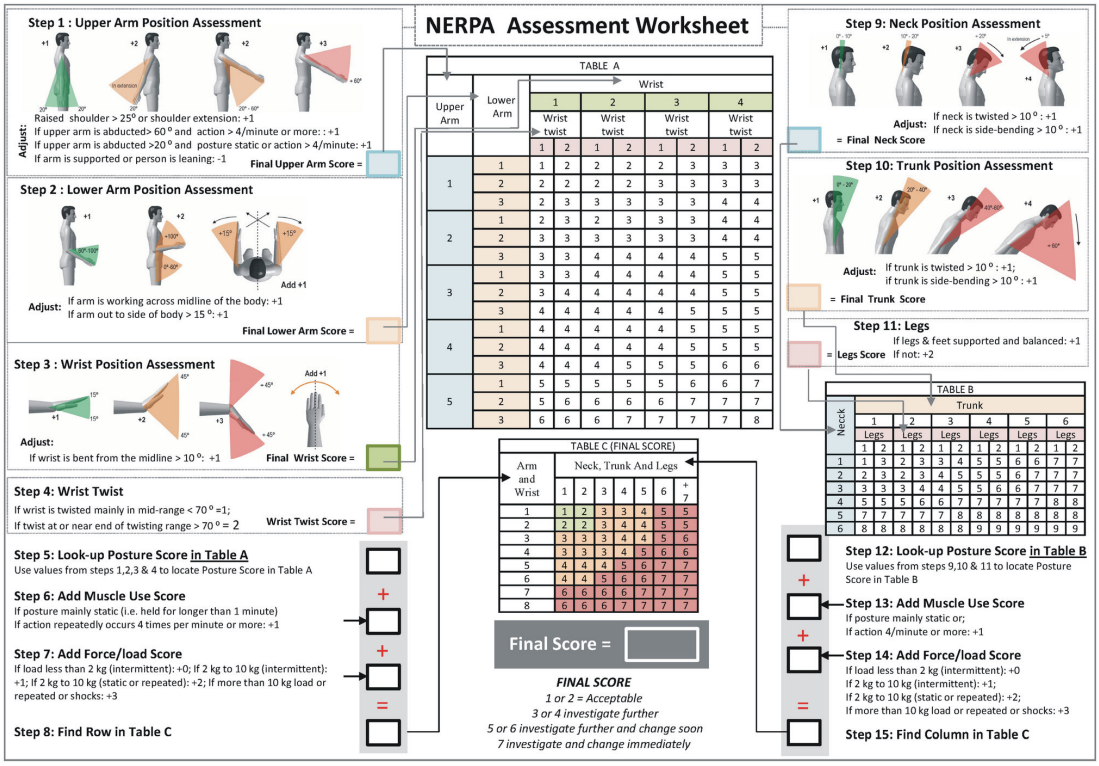
(a) The RULA worksheet.[9]

Figure 1.5: Examples of ergonomic assessment worksheets.

to accommodate the more unstructured and flexible sequence of tasks throughout a work day in carpentry compared to manufacturing work where ergonomic assessment tools are most often applied. The assessment carried out using this method takes its starting point in a range of postures and repetitive movements that are likely to occur in a carpentry tasks(Bhattacharya et al., 1997).

Many of the tools are worksheets that are typically used by an observing evaluator who, based on priorly gathered information, determines some key situations/positions to examine. The evaluator then combines observations with gathered information and follows the guidelines of the tool in order to calculate a score, an index or some other descriptive value. The benefit of such tools is usually that they are fast and easy to utilise and do not require evaluators with an advanced skill level or complicated technical setups. On the other hand the assessment is very much dependent on the evaluator's personal ability to estimate certain values such as angles.

In the recent years investigations into digital utilisation of the observational methods in combination with vision technology has given rise to a merged fourth category, one could say. (Vignais et al., 2013; Busch et al., 2017; Abobakr et al., 2017) Often the assessment tool digitalised is subject to some modifications. Some tools are hard to implement in a system as they have input parameters that are difficult to obtain through sensing techniques. An example hereof is the parameter coupling, i.e. grasp, in the NIOSH lifting equation. Other tools are easier to implement and have already been applied in automatic systems as they are dependent



(b) The NERPA worksheet.[10]

Figure 1.5: Examples of ergonomic assessment worksheets.

on parameters which more easily can be estimated using existing sensor technology. RULA is an example of such a tool which has been frequently implemented in systems striving to conduct automatic ergonomic assessment(Vignais et al., 2013). While RULA is subject to critique because no epidemiological data supports the patterns of influence resulting from combination of factors RULA implies, and because RULA does not consider how the risk for MSDs is influenced by the cumulative time spend in different ranges (Vignais et al., 2013); a recent study indicated that the tool outperforms other similar tools (Yazdanirad et al., 2018).

The rise of this new category and the interests in utilising ergonomic assessments as an integratedx part of systems promotes investigation into the potential, pros and cons of the different assessment categories from a technical point of view. In order to give an overview, the three categories of assessment methods have been assessed in respect to seven aspects. A scheme intended to map-out and display these properties is presented in Table 1.1.
From a technical point of view it is clear that the observational methods offer the most advantages and potential for utilisation in systems that are to assess realtime in a real work setting. This presumably is also the reason that mainly observational methods, such as RULA, have been incorporated in digital systems for such purposes. However, observational methods still have an important limitation in respect to technical incorporation which is related to being optimisable. This will be addressed in section 1.4.3.

1.4.3 Ergonomically Optimising Robot Movement

As mentioned in section 1.4.2 observational ergonomic assessment methods have been rekindled through the incorporation of digital adaptions with recent technologies. A natural question arising as an extension of these tendencies is whether observational ergonomic assessment meth-

Table 1.1: Dependencies, properties and potential of the assessment methods

	User input	Subjective	High accuracy	Invasive	Computationally demanding	Realtime	Optimisable
Self reporting	x	x	-	-	-	(x)	-
Observational method	(-)	-	(x)	(-)	(x)	x	(-)
Direct measurement	(-)	-	x	x	x	-	x

An overview of dependencies, properties and potential of the three categories of assessment methods from a technical point of view. x = yes or true, - = no or false. Parentheses indicate yes or no with reservations or somewhat true or false.

ods can be utilised for optimisation of system behaviour as opposed to mere evaluation. More specifically, the interest point in respect to this project is whether a robot could use the output of ergonomic assessment as an input and adjust its behaviour accordingly such that the exposure of a human collaborator to ergonomic risk factors is minimised. This, however, is challenging, and the works in literature addressing this are relative few. While some tools, like RULA, can be implemented in software it does not offer a direct, unique mapping between the score obtained and the pose analysed. Also, it is, in its original form, not differentiable nor continuous, which makes it difficult to use as a feedback measure for a system meant to optimise the pose. In literature ergonomics are used during design, interaction design or development of new methods in relation to robots interacting with humans (Park et al., 2004; Courreges et al., 2009; Crainic and Preitl, 2015; Wang et al., 2019). There are also examples of utilising ergonomics in relation to robotic behaviour towards humans (Bortot et al., 2013).

A few articles from Busch et al. (2017) describe an investigation of planning of motion and tasks in human-robot collaboration in such a way that the ergonomics of the human is optimised. In the work described, they fit a differentiable model to the REBA score and use this as a cost function for planning of motion and tasks in the system (Busch et al., 2017, 2018). This very similar to the system suggested and investigated in this project and the wanted properties of the system, but there are some points to address. Firstly, the system presented by Busch et al. (2017) uses marker-based tracking, where a non-invasive method might be more desirable because it is less obstructive. Secondly, the tests conducted on their system did indeed affect the REBA scores obtained positively; however, the consequences of the method were not addressed. By minimising and thus potentially eliminating some of the postures with high risk scores without taking into account the time spend in those poses, the resulting postures and movements after applying the method might be characterised by higher repetition, longer durations and less variation. This relates to the third critique point which is that their method in fact does not take time spend in postures or repetition of movements into account.

Peternel et al. (2017) also present a method for optimising ergonomics, i.e. minimising mechanical overloading in the human body during human-robot collaboration though adaptive robotic movements. In their work they build their method around forces and overloading joint torques rather than utilising known ergonomic tools. Their method is to minimise overloading

joint torques and maximise manipulability of the human arm. The overloading joint torques are found through the variations in the Center of Pressure (CoP), which are estimated using a dynamic model of the human whole-body CoP integrates external load and interaction forces from robot end-effector measurements. Their method was tested on movements in the sagittal plane and with a human model including 5 Degrees of Freedom (DOF). During testing the body kinematics were monitored using a MVN Bodymech suit (Xsens Technologies), a IMU-suit and the subjects muscle activity was monitored using Electromyography (EMG). The muscle activity measured through EMG showed a successful decrease in the overloading joint torques compared to the unoptimised condition. (Peternel et al., 2017)

In literature there are also examples of research in adapting robot behaviour and human-robot collaboration to ergonomics on a higher level than only robotic movements. One example is the work already mentioned by Busch et al. (2018) where they strive to optimise task allocation and motion planning simultaneously for generating more ergonomic situations in human-robot collaboration. Another example is the work by Pearce et al. (2018) in which they attempt to create a framework for optimising task assignment and scheduling in human-robot collaboration based on time and ergonomics. Their method presents an approach for scheduling tasks based on a set tradeoff between minimising makespan of the task and the strain of the work elements on the human measured using a variation of the Strain Index (SI) presented by Moore and Garg (1995).

Though the more recent research in the area is more advanced, researchers have previously speculated in how robots can facilitate ergonomic conditions in Human-Robot Interaction (HRI). For example, Lynch and Liu (2000) investigated the opportunities in create guides for the support of manually manipulating heavy loads from one state to a goal state. The guide shape was created according to a set of parameters, among other an object function based on ergonomics. They also speculated in using a cobot for implementing the guides.

The field is advancing rapidly and within the last one and a half year several new works in the area of ergonomic adaptation of robotic movements in HRI have been published. The work presented by Peternel et al. (2017) has been advanced by groups consisting of some new researchers as well as some of the original researchers on the project(Lorenzini et al., 2019; Kim et al., 2019). The advancements presented include an overloading joint fatigue model and a recovery model created in order to consider commutative effects of overloading joint torques (Lorenzini et al., 2019). Other advancement presented are the combining of the previous mentioned advancements with a framework for adaptation to intentions (handedness, intended manipulation, change in human position in the workspace) and the change from using IMUs for online monitoring of the human kinematics to the use of a stereo-vision camera together with OpenPose (Kim et al., 2019).

Shafti et al. (2019) presents a different approach utilising an ergonomic assessment tool. Focussing only on the ergonomics of the arm they use the parts of RULA that describe the arm. In their approach a human is monitored using an RGB-D camera, the Microsoft Kinect, and IMUs to obtain the RULA score of the arm. If the observed score is higher than a set threshold the robot will assist in moving to a more ergonomic state by adjusting the position of the workpiece it is holding. The RULA score for the arm is determined by the states of different arm segments. By monitoring the individual RULA scores of the arm segments, the arm segment causing a high RULA arm score can be determined. For each of the states of the arm segments Shafti et al. (2019) have determined an appropriate robot response, and using the individual RULA scores and a prioritisation protocol the appropriate response for improving an unergonomic state can be found. The presented method was verified through testing that showed that the method indeed did result in better RULA arm scores than without robot assistance over the course of a conducted test task.

Chapter 2

Project Delimitation

This chapter specifies and outlines the focus of this project. Also, requirements for the desired system are specified

2.1 Scope of the Project

This section serves to summarise and outline the problem area investigated in this project.

In the previous chapter it has been addressed what challenges the meat packing industry is facing, and how improvement in the working environment of the butchers is wanted. A specific work package of the ACMP project was introduced as the context, and a general system outline and an exploration of state of the art of related areas was provided. The general system outline was simply concerned with a system capable of adapting robot movements to ergonomic conditions of the human. However, if a system with these capabilities is to be implemented for a specific application, it is likely that there are some constrains to consider in relation to the specific context in which it is applied. The butchers working in meat production are making rapid movements with their upper limbs in particular with a high number of repetition. Some of the pose estimation methods explored in section 1.4.1 have the benefit of not being constrained to a small dedicated area and might work wireless at the same time, but they, on the other hand, are dependent on devices mounted on the human or special garments. These are characteristics which are not convenient in the context of the butchers, who need to wear protective equipment, and while the ability to work in an unconstrained space is attractive, it is not strictly needed as the butchers typically work in a limited confined space. Similarly, some of the methods might have a high precision but be dependent on strict environment configurations and markers on the body, which, again, is inconvenient for the hard working butchers. Based on these considerations it is deducted that a "non-invasive" method would be preferable. Consequently, this leads to a focus on observational methods using vision technology without markers.

In section 1.4.3 recently suggested methods for ergonomic adaptation of robotic movements was presented. Regarding the method by Busch et al. (2017), it would be interesting to examine their exact methods performance in respect to ergonomic improvements when utilising marker-less vision-based pose estimation. This will, however, not be the focus of this project; nonetheless 2D marker-less vision-based pose estimation will be utilised. Also, it is common for ergonomic assessment methods to mainly consider movements in the 2D sagittal plane (see Figure 2.1) (Roman-Liu, 2014). This project will seek to advance the field of ergonomic adaptation of robotic movements by including some element considering the repetitiveness, duration or variation in the adaptation approach, which is not frequently encountered in the literature. The investigations will be conducted in the context of a specific use case which will be described

in section 2.1.2.

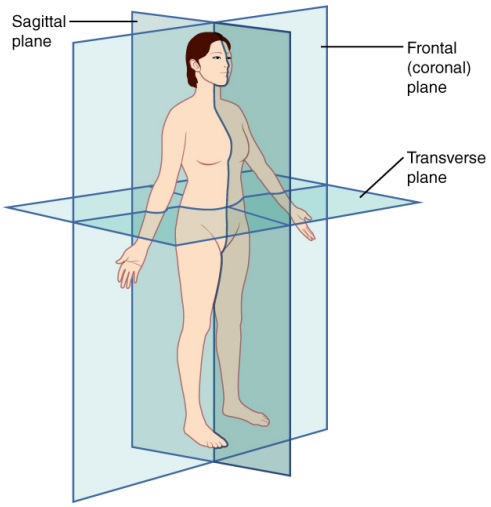


Figure 2.1: The three most commonly used planes of the body.[1]

In relation to collaborative robots as the one that is included in the proposed system in this project, safety is always a topic that is of great concern. Although this project might address safety in relation to the parts implemented in this project and in relation to any user testing, it is not considered a focus point of this project. No in depth investigation of safety issues in relation to collaborative robots will be conducted, nor will the system be implemented with excessive software or hardware modules to ensure an entirely safe system as would be required in a commercial context.

2.1.1 Hypothesis

Previous works have proven that robotic movements can be adapted according to ergonomic measures in order to minimise ergonomic risk measures, However, there are little work shoving that it is possible to do so while actively counteracting repetitiveness in the operations of the human. The hypothesis for this project is as follows:

Through the use of ergonomic measures and constrains it is possible to optimise the ergonomic state of a human collaborating with a robot as well as minimise exposure to repetitiveness.

The aim of this project is, thus, to examine how and whether a collaborative robotic system which adapts according to the operators ergonomic conditions can be implemented in such a way that it considers repetitiveness.

2.1.2 Project Use Case

This section describes the use case in the context of which the research of this project will be conducted.

[1] Source: Donna Browne, (Browne, 2015)

As stated earlier, the context of this project is the second work package of the ACMP project. The specific use case for the investigations in this project is inspired by the mentioned work package. The use case is thus the handling and presentation by the robot of belly pieces to the butcher for processing purposes. The processing operation assumed here is cutting of the edges of the belly pieces in order to get a wanted square shape of meat. The cutting of the meat is standing work for the butcher.

In practice, the use case chosen will be simulated through simplified setups for convenience of the development in this project. Likewise, no actual meat pieces will be used during this project.

In a potential application setting of a final system it is likely that more people will appear on input images from the sensors. In order for a robot to be able to work in collaboration with and adapt to the right human operator, the system in that case should have some method in place for distinguishing between the human collaborator and other humans in view and thus prioritise adaptation to that collaborator above others. Nonetheless, the use case and test setup in this project will be constrained to having one person in the image view at all times; thus, a prioritisation method will not be necessary nor will it be implemented.

2.2 Requirement Specification

This section describes the functional requirements of the system to be developed. The requirements are justified with arguments about relevance and importance. Each requirement will be described in its dedicated paragraph.

Vision system The system needs a means of monitoring the human operator. Since the system is developed with the intent of longterm monitoring with the purpose of providing estimated poses as input to a robot, it is inconvenient to have a system dependent on elements or units that are physically attached to the monitored worker. Consequently, it is preferable to have a non-invasive vision-based system able to observe the worker analytically without the use of markers. Therefore, a vision system able to detect human poses is a requirement.

Ergonomic Assessment As the system is supposed to adapt based on ergonomics, a method for assessing the ergonomic conditions is required. The method should include a measure for posture which is commonly used as the central factor in ergonomic assessment methods. Furthermore, the method should include a measure for repetition as required by the hypothesis stated in section 2.1.1.

Adaptive Behaviour The system should be able to adjust its movements and/or configuration based on input in order to facilitate optimisation of the ergonomics conditions of the human pose and minimise exposure to ergonomic risk factors, including repetitiveness.

2D capabilities The system should be able to detect human's exact pose in the 2D sagittal plane with an accuracy of ±2.5cm but also be able to make adaptive movements in the equivalent 2D Cartesian space.

Realtime As the system is supposed to continuously adapt to a human present in the work space, realtime properties are necessary. Due to the slow speed the robot should adjust with, the realtime constraints in relation to this project are less restrictive than in other contexts. From research within prosthetics it is known that the recommended delays within the 100-400 ms interval are unnoticeable for the human (Farrell and Weir, 2007). In the context of this project,

it is not required that the human does not perceive the delay. However, the delay should also not be too large such that the adaptive moves when executed are outdated and improper for the current situation. The robot should during execution of an action receive control commands with a rate of at least 2 commands pr. sec., and the delay in response should ideally not exceed 1 sec. such that the human operator in spite of a small delay still has the notion of immediate response.

Collaborative robot manipulator Although safety is not a focus point in this project it is still important to include in considerations; especially, as user testing might be conducted and subjects should be safe. As a consequence, it is a requirement that the system is developed utilising a designated collaborative robot manipulator which is designed to work in close proximity to humans. Such manipulators are characterised by force, torque and/or speed limits, rounded edges, and often also soft padding and elasticity in joints.

These are the system requirements that have been identified initially. As the work proceeds new issues encountered might introduce new requirements or lead to modifications of existing requirements.

Chapter 3

Methods and Means

This chapter investigates some of the methods relevant to this project and to the desired system described in chapter 2.

3.1 Non-invasive Vision-Based Human Pose Estimation Methods

As a result of the delimitation to non-invasive vision-based pose estimation methods, this section explores a variety of such available methods. A requirement to the method needed is that it outputs key points or a "skeleton" model of the human and, e.g., does not simply output a segmented point cloud of the human. Furthermore, the method should be a more or less off the shelf solution since this is not the central focus point of this project. As a consequence, only solutions with available software or code are investigated.

One available method is PoseNet by Tensorflow. PoseNet is able to detect poses of one or multiple persons in images. An image is first fed through a convolutional neural network which then can return different confidence measures, such as overall pose confidence score, and key point information, up to 17 key points per pose. PoseNet has some limitation. First of all, the output model is inconvenient for determination of, e.g., the spine. Second, the method is only trained for pose detection in 2D input. This software is open source. (Papandreou et al., 2017) (Tensorflow, 2018)

HRnet is a different pose estimation network presented by Microsoft in collaboration with University of Science and Technology of China. This network utilising high resolution representations throughout has performed very well breaking three records on the COCO dataset. The network structure consists of parallel pipelines of convolution operations which are interconnected with up and down samplings between the pipelines. (Sun et al., 2019)

Another available open source method is the OpenPose developed by CMU perceptual computing lab. This software is able to detect multiple humans, it can run both on the CPU and GPUs and it can be used to detect key points and estimate poses realtime in both 2D and 3D. The software also enables the use of different pose models. (CMU Perceptual Computing Lab, 2019; Cao et al., 2017)

wrnchAI is a network by wrnch with a comparable performance in respect to precision and recall to OpenPose. OpenPose outperforms wrnchAI in classification of large images while the opposite is true for small input images. The wrnchAI, however, is much faster. The software is available through access request by signing up on wrnch's homepage. (Gupta, 2019)

The Microsoft Kinect camera mentioned in section 1.4.1 is also a non-invasive method that could be considered. The device, which is a stereo camera, can be used with the connected software developed by Microsoft or with software from other parties such as software from iPi soft

(iPi Soft, 2019). The Kinect cameras are also able to provide positions of key points in 3D space. Unfortunately there are also reports that the Kinect system fail to successfully detect the pose if there is any self-occlusion or if other object occlude parts of the human(Obdrzalek et al., 2012).

In addition to the methods mentioned there are a selection of methods presented in research papers for which the code is also provided; however, as it is the intention to integrate the method into a Robot Operating System (ROS) application it is a requirement that a ROS interface or wrapper is available for the method used as it will be too time-consuming in respect to this project to create one. In general the less known methods do not have ROS wrappers available as those are often made by external developers taking an interest into certain methods (often methods with a lot of publicity). Of the methods mentioned only PoseNet, OpenPose and the Kinect are solutions that can easily be interfaced with ROS. Due to the need for an interface for ROS, the flexibility of choosing between models, and the reports on instability of pose detection by the Kinect, OpenPose was chosen as the method to be applied and investigated initially.

3.2 Ergonomic Feedback

Currently there seem to be very few ergonomic assessment methods available that would serve as a suitable tool for ergonomic feedback for robot control and which have been designed for the purpose. The only methods encountered in literature at the onset of this project are two methods already mentioned in section 1.4.3. The first one is the differential REBA by Busch et al. (2017), which essentially is a modification of an existing tool for a different purpose (pure evaluation) and which has been subject to quite some critique. The second is the method striving to minimise overloading joint torques described by Peternel et al. (2017) which to some extend assumes equal severeness of torques in all joints.

Perhaps the field of ergonomic human-robot interaction lacks a model or a function that is purposely created to aid ergonomic optimisation. Yet, designing such a function is a study in it self; one that requires extensive knowledge about the musculoskeletal system and ergonomics and as well as control. Design of such a function is beyond the scope of this project. Consequently, the model used for ergonomic optimisation in this project will either be a modification of an existing ergonomic assessment method or a constructed model based on well known ergonomic facts and correlations.

3.3 Structure of the Work

Due to the rather explorative nature of this project, the work is structured based on principles from agile management methods. The benefit of the agile methods is the continuous adjustment of the work to changing conditions to limit redundancy and inefficiency.

There are several examples of well defined agile methods such as The Lean Start-up or Scrum; however, while the work will utilise some of the general principles common in agile methods, no established agile method will be strictly applied. A core aspect in many agile methods is the iterative development approach which gives opportunity to advance the work iteratively. This is usually done continuously through small cycles of work and evaluation of the achieved and planning the next step. The work in this project is also conducted in iteratively. An Initial System (IS) will be implemented and subsequently advanced in the following iterations. The work of each iteration will be evaluated and adjustments to the direction of the work and, if necessary, the specified requirements will be carried out.

Due to the fact that the implementation work is carried out utilising aspects of the agile method,

the chapters describing the implementations will be characterised hereby. The first implementation chapter will be about the implementation of the IS while the subsequent chapters will be about iterations of advancement.

Chapter 4

Design of the Initial System (IS)

The initial development work was conducted on implementing an IS. In this project the IS is an elementary version of the full system going from a visual input obtained from observation of a human collaborator to an ergonomic adapted robotic movement. In the following sections the system structure of the IS and the development work will be described.

4.1 System Overview

The design of the system implemented as the IS consists of three main components: a camera, a control system running on a desktop and a robot. The camera delivers input which is processed and used for generating control output for the robot (see Figure 4.1).

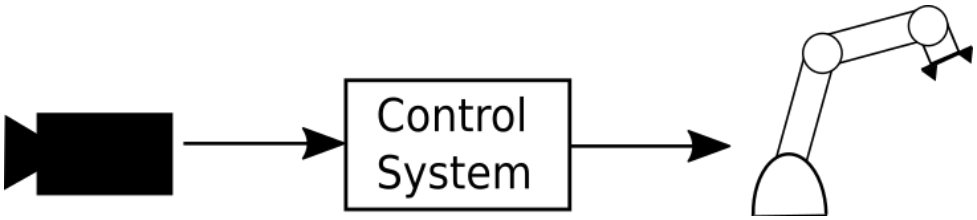


Figure 4.1: The overall system in the IS

The system is meant for working in close relation to human operators and adapting to them. As a consequence the system could be loosely viewed as a closed loop system when including the human in the loop (Figure 4.2). The "feedback loop" is given by the camera observing the human and builds on the assumption that the robot movements will affect the human's pose.

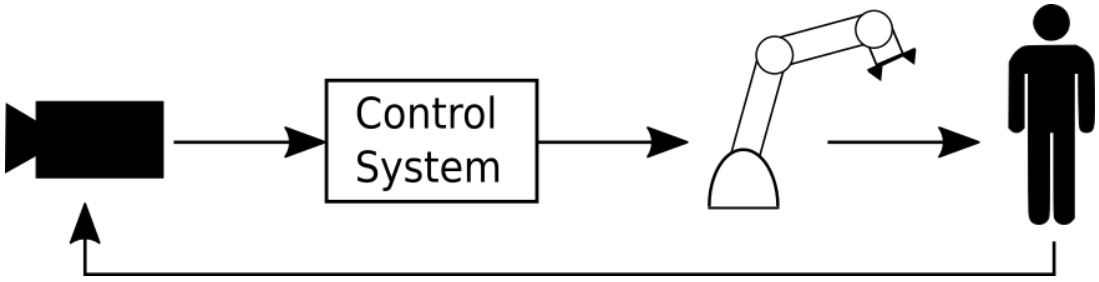


Figure 4.2: The closed loop system principle including the human.[1]

[1]Source: van den Broek and Moeslund (2020)

The IS in this project is intended to be a basic first version of a fully functional system, meaning a system adapting to ergonomic conditions of a human based on visual input. For simplicity reasons, the task addressed in the IS in respect to the use case described in section 2.1.2 is reduced to adapting only the height of a presented belly piece. This can be viewed as equivalent to adjusting the height of a work surface. In the IS the inclusion of a repetitive measure will not yet be considered. The adaptation of robotic movements will in the IS happen based on a well known ergonomic fact as mentioned in section 3.2. This fact relates to the ergonomic risk factor posture and is the well known fact that standing upright with a straight back is preferable over any over stretching of the back or stooping posture for standing work in any extended period of time. More concretely, the idea is that the system observes a single angle in the body pose, the hip angle, by capturing from a side view of the human with an RGB camera. In this way, the human pose in the 2D sagittal plane (see Figure 2.1) is observed. This also complies with the movements mainly considered in ergonomic assessment methods as mentioned in section 2.1. The assumption is that the hip angle in standing work will indicate whether a butcher is over stretching or stooping.

For different states of the observed hip angle, appropriate robot responses should be specified which can be executed whenever the different angle states are observed. Based on the hip angle observed the system should either be static, elevate or lower the work surface in order to facilitate correction of the unergonomic posture and guide the butcher towards an upright working posture. Thus, the IS only has one degree of freedom. The detailed implementation and functionalities of the system will be elaborated in later sections.

The system described is achieved using some "out-of-the-box" solutions, software implementations and existing hardware that suffice for the purpose of this initial demonstrator of a system. The following sections will describe the hardware and software used for the IS.

4.1.1 Hardware

This section will describe the camera, the desktop and the robot used.

The camera used is a Logitech$^{\text{ő}}$ QuickCam E 3500, which is a standard webcam. The camera has true resolution of 640×480 pixels and a hardware frame rate limit of 30 fps. The camera has a diagonal Field of View (FOV) angle of 60°. The camera connects with a corded usb 2.0 connection. Logitech (2019)

Figure 4.3: The Logitech$^{\text{ő}}$ QuickCam E 3500 used for the IS.[2]

[2]Source: Logitech$^{\text{ő}}$, https://secure.logitech.com/assets/30973/e3500-front-and-side.jpg

The desktop computer used has an Intel Core i7-3770 CPU with 32 GB memory running with 4 cores at 3.40 GHz. The computer is equipped with an Nvidia GeForce GTX 1080 Ti graphics-card with 3584 Cuda cores and a total memory of 11 GB (Nvidia, 2020). The desktop is running Ubuntu 18.04.2 LTS (Bionic Beaver).

Figure 4.4: The Sawyer collaborative robot from Rethink Robotics used for the IS.[3]

The robot used for the IS is the collaborative robot Sawyer from the company Rethink Robotics (see Figure 4.4). The robot is referred to as collaborative due to its rounded edges, soft padding, force limitations, elastic actuators and user friendly software, which makes it safe to use in close proximity to humans and easy to operate and apply (Rethink Robotics, 2019). The robot has a manipulator with 7 Degrees of Freedom (DOF), a payload of 4 kg and a max reach of 1260 mm. Furthermore, the robot contains a screen and two cameras, one placed just above the screen and one placed close to the end effector. Connecting to the robot happens through ethernet/IP. The manipulator and controller is mounted on a pedestal with wheels making it easy to move around, and with supportive feet enabling stable placing.

4.1.2 Software

In relation to the implementation of the IS some choices have been made regarding which software to use. The part of the system that interacts with the robot is beneficial to program in Robot Operating System (ROS) as it is a flexible environment with a large community enabling easy implementation of robot applications. ROS comes in different versions, but one of the most used, and thus most supported in extra libraries, is the ROS Kinetic Kame. Therefore, this ROS distribution is the one chosen for the IS.

The ROS only runs on Linux and since the Kinetic distribution is primarily targeted Ubuntu 16.04 (Xenial) this release is used.

In the IS an existing human pose estimation library is used. Due to accessibility, the arguments provided in section 3.1 and previous experience with the software, the OpenPose[4]software from CMU Perceptual Computing Lab is chosen.

The OpenPose is dependent on parallelism and GPU computing in order to achieve a realtime performance. Therefore, Nvidia's CUDA and cuDNN are needed for OpenPose to run satisfactory. CUDA 10.1 and cuDNN 7.0 are used in the IS.

[3]Source: RG Robotics, https://www.rg-robotics.com/wp-content/uploads/2018/09/collaborative-robot-sawyer.png

[4]https://github.com/CMU-Perceptual-Computing-Lab/openpose/

For ROS to be able to communicate and interface with OpenPose in an appropriate and convenient way a ROS wrapper for OpenPose was needed. Luckily, different developers have implemented their suggestion and posted it opensource on git for free use. Kevin Zhang has made a convenient wrapper available on which he is still exercises support continuously making sure it works with the last commit on OpenPose.[5]

Finally, Rethink Robotics provides a software, Intera SDK[6], which serves as a bridge between the Sawyer robot and ROS. This is used for parsing control information from the robot application implemented in ROS to the robot's own system.

4.2 Reproducibility

It is essential to the quality and validity of scientific work that the results achieved are reproducible. In order to ensure and facilitate reproducibility of the system implemented during this project, the system is implemented in a virtual environment using Docker. Docker is a useful, valuable tool for ensuring fast reconfiguration of software setups on different computers. As Docker was an unfamiliar concept, time was invested in getting to know Docker and its usage. In the following an insight into how Docker works and how it was used during this project is given.

4.2.1 Docker

Docker is a programme used to create virtual environments in a simple and flexible way. The environments are set up using the so called "Dockerfile". The Dockerfile serves as a recipe for a certain configuration of a virtual environment. In the Dockerfile the base system along with the necessary subsequent commands to set up the system exactly as wanted are specified. From the Dockerfile a "blue-print" called a "Docker Image" is build. Once built, images take up a considerable amount of storage space; nonetheless, with an image available, an instance of a system with the configuration as outlined in the image can be created in the form if a Docker container within seconds. This also makes it easy to bring configurations from one device to another as long as they both have Docker set up. Building the image is a time-consuming process, and so can the scripting of the Dockerfiles be. Furthermore, Docker also provides an online community which facilitates the sharing of created Docker images. When a virtual environment of a certain configuration is needed, it can either be downloaded from the online platform "Docker hub" where official images and unassociated developers' personally implemented images are available, or it can be scripted in a Dockerfile and build. Due to the fact that there was no Docker image found on Docker Hub with the exact system configuration needed for this project, a Dockerfile was scripted.

4.2.2 The Dockerfile

Initially different available images were investigated. The system needed is one with ROS and OpenPose installed, but a Docker image containing this configuration was not available on Docker Hub. ROS has official Docker images available with different ROS versions, but they do not contain OpenPose. Similarly, some developers have made Docker images available containing OpenPose but not ROS. Some of the OpenPose Docker images were even deprecated as they were downloading the latest version of OpenPose code but did not contain the functionalities for handling the latest version resulting in a multitude of errors. Based on these facts it was decided that a Dockerfile of a system containing the necessary parts in a time invariant way

[5] https://github.com/firephinx/openpose_ros
[6] https://github.com/RethinkRobotics/intera_sdk

would be scripted. Also, scripting of Dockerfiles is useful to learn for future projects as well.

The process of creating the Dockerfile along with the challenges encountered in the process and how these were overcome is described in Appendix A. The Dockerfile for the IS contains the majority of the softwares mentioned in section 4.1.2; however, scripts coded during this project which undergo continuous change are included as linked directories from the host sytem when a container is run. More details as to why this choice was made can be found in Appendix A. The final Docker image has been shared on Docker Hub[7] as it might be valuable to others, and likewise, the the Dockerfile has been shared in a public git repository om Bitbucket[8].

4.3 Connection Between the Docker Container and Sawyer

Once the wanted system with the needed elements had been setup in Docker, the next step was to connect the running Docker container with the robot.

The robot is connected to the desktop through an Ethernet connection on a shared LAN. In order for the Docker container to be able to communicate with the robot, the Docker container should be able to access the LAN and the communication on the network.

Docker provides different flags that can be used when running the containers, one of them is the `-net` flag which, when given the argument "host", enables the container to access the host network. The container should be able to access the LAN that the "host" desktop is a part of. However, when running the container, and running the script provided by Intera SDK for establishing connection to the Sawyer robot in ROS it was not possible to communicate with the robot properly. The issue was solved through the use of the `-add-host` when running the container which made it possible to map the Sawyer host name to its IP address and thus enable successful two-way communication between the robot and the desktop. The solution was found through an explorative process; for details, please see Appendix B.

4.4 The ROS Setup

This section describes the robot application implemented in ROS and its structure.

The structure is inspired by, and very similar to, the control system overview described in section 1.3. The robot application implemented in the IS consist of several ROS nodes, some provided by other developers which already existed prior to this project, and some specifically programmed for this project. A list of the ROS nodes used in this application and a brief explanation of their functionality is provided below.

[7] https://hub.docker.com/r/mkvdb/ros_openpose

[8] https://bitbucket.org/mkvdb/mkvdb-open-source/src/master/dockerfiles/ROS_Openpose_Image/Dockerfile

cv_camera_node A node responsible for obtaining the raw image from the USB camera and publishing it in a ROS topic.

openpose_ros_node A node which subscribes to a image feed topic and uses the OpenPose software to detect human poses in the images and, subsequently, publish the pose information in a ROS topic.

angle_calculation_node A node which subscribes to the pose information topic, calculates relevant angles between segments of the body model and publishes this information.

robot_adaptation_node A node that makes decisions about adaptive moves based on received angle information and publishes control commands.

robot_control_node A node parsing control commands published to a specific topic through to the robot.

An overview of the nodes and their intercommunication can be seen in Figure 4.5.

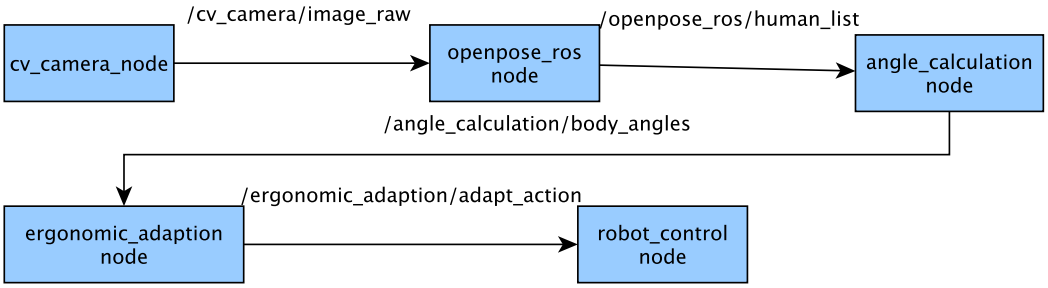


Figure 4.5: The structure of the robot application with the communication ways implemented in ROS. The names of published topics are provided along the arrows.

In the following sections the functionalities will be elaborated and the use and implementational work, if any, explained for all of the mentioned nodes except the simple cv_camera_node.

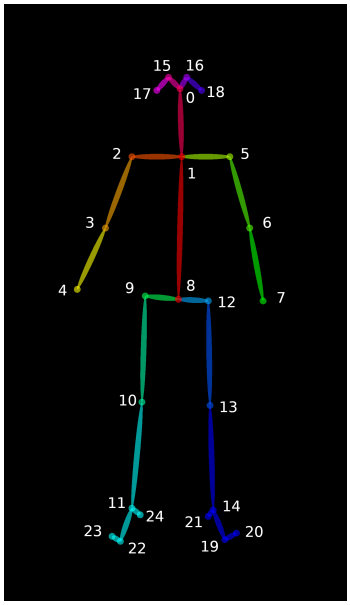
4.5 The OpenPose ROS Node

This node is the OpenPose-ROS wrapper. It subscribes to any topic in which image frames are published. In theory you can let OpenPose process images of any visualised modality (RGB, thermal, depth), but since the network at the core of OpenPose is trained for RGB images specifically, this is the modality that will work the most robustly. In the implemented system the node subscribes to the /cv_camera/image_raw topic posed by the cv_camera_node based on the input from the Logitech camera described in section 4.1. The images posted to the topic are RGB images. When an image is published to the topic the openpose_ros_node analyses the image by detecting humans in the image and estimating the body model of each of the humans in the image. OpenPose is able to use and fit one of a few different models to the detected humans. The model used in the IS was the BODY_25 model, which describes the pose through 25 key points throughout the body (see Figure 4.6a.

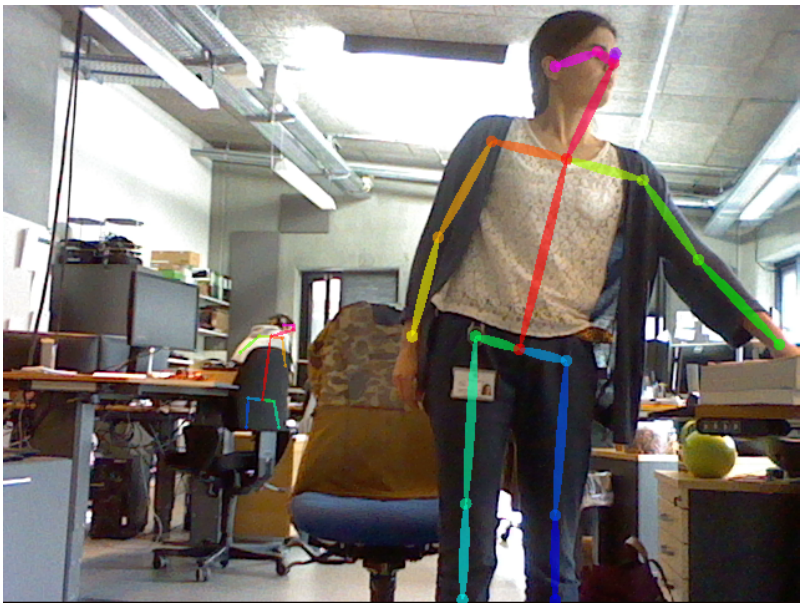
OpenPose works with a bottom-up approach. The core of the method is the use of a Convolutional Neural Network CNN and Part Affinity Fields (PAFs), which are sets of 2D vector fields that encode the location and orientation of limbs. In OpenPose a full image, possibly depicting

several people, is fed as input to a multi-stage CNN which jointly predicts both confidence maps and PAFs. The confidence maps are used for body part detection, while the PAFs are used for part association. In a subsequent step body part candidates are associated through a set of bi-partite graph matchings. The found pairs from the matchings are in the end combined such that a full body pose for each person in the image is obtained. Finally, the key points for the body pose can be provided according to the chosen body model. (CMU Perceptual Computing Lab, 2019)

In Figure 4.6b OpenPose can be seen in action detection key points of humans in an image captured by the Logitech[ő] QuickCam using the BODY_25 model.



(a) The BODY_25 pose model used by OpenPose in the IS.[9]



(b) An example of how the key points are detected in an image using the BODY_25 model.

Figure 4.6: The key points found by OpenPose.

The node publishes its output to the /openpose_ros/human_list topic. The output is published in a format which contains header information of the published message self, header information about the image frame processed, number of humans in the frame, and a list of information from each human in the frame. Each entry in the list of humans will contain several parameters and selections of key points related to the human in question, among other the key points of the body pose.

4.6 The Angle Calculation Node

This node serves as the ergonomic estimation module of the IS. As mentioned in section 4.1, this first version of the system is to adapt based on a simple ergonomic posture measure, namely, the hip angle. The context of the system is work conducted standing upright; thus, if the angle at the hip is changing from the standard upright position, the observed human must be either over stretching the back or stooping, both of which are unergonomic working positions for an extended period of time. Hence, this node estimates the hip angle which serves as the ergonomic assessment model in the IS.

[9]Source: CMU Perceptual Computing Lab, https://github.com/CMU-Perceptual-Computing-Lab/openpose/blob/master/doc/output.md

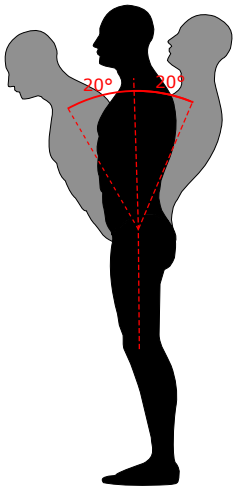
The node subscribes to the /openpose_ros/human_list topic published by the openpose_ros_node. From the data it receives it currently uses the number of humans in frame parameter and the body pose information from each of the detected humans. The hip angle is calculated based on segments of the body model which spans between pairs of key points as shown by the colourful lines in Figure 4.6a. The way the angle is calculated is based on vector calculations. As a result, the first step carried out by the angle_calculation_node when it receives input, is to check whether the necessary key points for each of the humans were detected and calculate the vectors if they were. The first vector calculated is the "spine vector" which represents the the spine and is calculated with the direction from the manubrium to the lower end of the spine. Also, vectors representing, respectively, the left and the right thigh or humerus are calculated with the direction pointing towards the knees. When the vectors are calculated in this way, the angle between the hip and the thigh angles will be close to zero when the human is standing upright regardless of the angle of observation.

If none of the thighs or the spine vector can be calculated, the node will print a warning to inform that insufficient key points were detected. If, however, both or only one thigh and the spine could be calculated the angle between the spine and the legs, the hip angle, will be calculated using either an average of the two thigh vectors or simply the single available thigh vector. Since the vectors are calculated based on only one RGB image, the angles calculated between them will be projections of the real angle in 3D space onto the image plane. As a consequence, the ideal observation angle or capturing angle for the camera will be directly from the side.

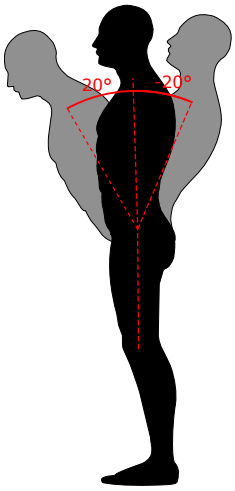
The hip angle was calculated using the formula for angles between two vectors which is shown in Equation 4.1.

$$\Theta_{\text{Between}} = \arccos\left(\frac{\vec{v} \cdot \vec{u}}{\|\vec{v}\| \, \|\vec{u}\|}\right) \tag{4.1}$$

The problem with this equation is that it calculates the angle *between* vectors and not from one vector to another; meaning, the calculated angle has no direction or sign (see Figure 4.7). In practice this means that it is not possible to distinguish whether a human observed from the side is bending towards the left or the right in the image. For the applications at hand this is not beneficial, because information about whether the human is over stretching or stooping is necessary in order to determine the right course of action.



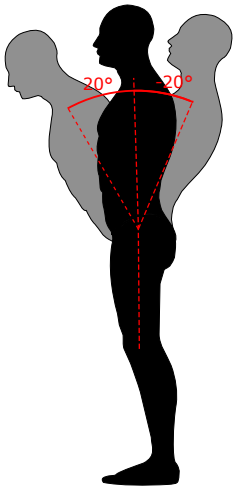
(a) The unsigned angles obtained using Equation 4.1



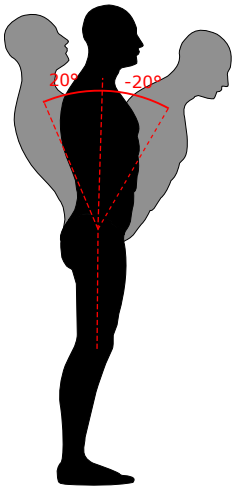
(b) The wanted behaviour with signed angles.

Figure 4.7: Examples of the angle calculation behaviour.[10]

In order to obtain a signed angle, the cross product of the two vectors is calculated. The sign of the crossproduct is then assigned to the calculated angle. This gives a sign based on the angle direction, but as shown in Figure 4.8 the sign of the angle simply follows the direction in the image and not the orientation of the human. It is beneficial if the angle directions is relative to the human pose such that the system is more flexible and less dependent on the capturing angle.



(a) The signed angles obtained by using the sign of the crossproduct.

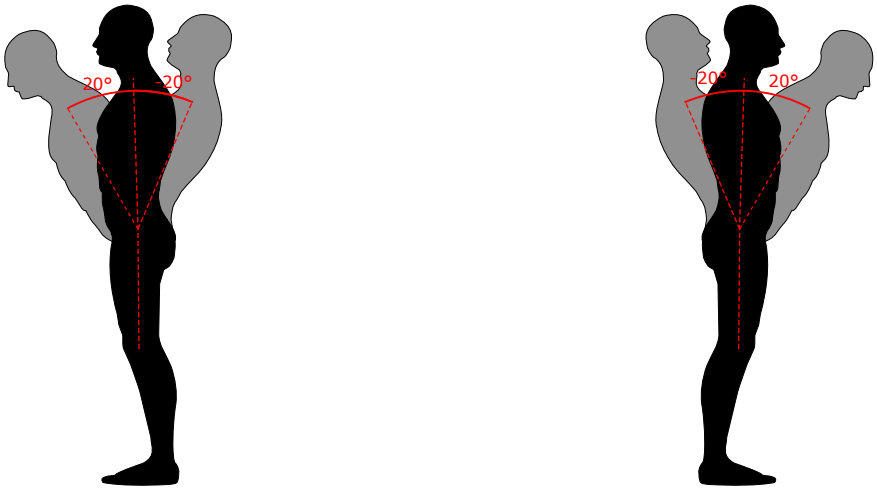


(b) The sign of the angle is consistent with the direction of the angle in the image plane.

Figure 4.8: Examples of the sign of the calculated angle dependent on the angle direction in the image plane.[11]

To be able to assign the sign in relation to the pose some additional vectors and angles have to be calculated. When the BODY_25 model is fit to an upright human, most pose segments will be fitted in a way such that they in a 3D space all would be, or nearly be, contained in one plane. This also makes it difficult to determine in which direction the detected human is looking. The only exceptions are some of the segments between the key points in the face and neck. The segment between point 0 and point 1 in Figure 4.6a goes between the manubrium and the nose, and it gives some indication of which direction the nose is pointing. This was utilised by calculating the vector from the nose to the manubrium and, subsequently, calculating the

signed angle from the spine vector to the nose-manubrium vector. The sign of this angle will indicate roughly in which direction in the image the human is oriented. Based on the obtained orientation of the human, a sign change will be added to the hip angle if necessary. The resulting behaviour can be seen in Figure 4.9.



(a) The signed angles.

(b) The wanted behaviour with signed angles following the orientation of the human.

Figure 4.9: Examples of the sign of the calculated angle dependent on the human orientation.[12]

Using the method described above a vector containing the signed hip angles for all humans in the image for which sufficient key points were detected is created and published to the /angle_calculation/body_angles topic.

4.7 The Robot Adaptation Node

This node is responsible for the decision making in regard to the adaptive moves the robot should make. This mapping between the input and the appropriate robot actions can be called the robot response model. In the IS the functionality of this node, and, thus, the robot response model, is quite simple. The robot_adaptation_node subscribes to the /angle_calculation/body_angles topic and uses the hip angle it obtains from the received message. Based on the angle, one of three actions is chosen by the robot response model and published to the /ergonomic_adaptation/adaptive_action topic. The decision by the robot response model is made using thresholds. The thresholds were found through a simple trial and error test where the fluctuation in the hip angle of a person standing upright with the side towards the camera was observed. The standing upright interval was initially found to be from $-10°$ to $30°$ but was later refined to be $-10°$ to $2°$ for an upright non-curved back.

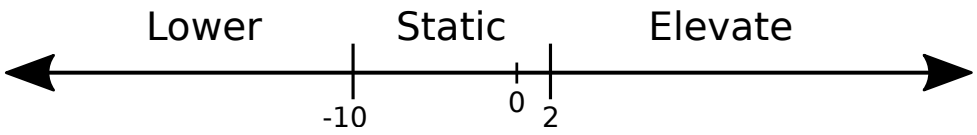


Figure 4.10: The correlation between the thresholds and intervals in the hip angle and the adaptive move chosen by the robot response model. The angle is reported in degrees.

[12]Figures with the side silhouette of a man are created based on image from http://archive.zbrushcentral.com/showthread.php?170747-Critiques-please!!

The correlation between the thresholds and the action chosen is as follows: if the hip angle is smaller than −10°, the action chosen will be lowering the work surface; if the hip angle is in the interval from −10° to 2° the action will be to hold the work surface static; if the hip angle is larger than 2° the work surface will be elevated (see Figure 4.10).

As an extra division the angle interval for the elevation action has been divided into two angle intervals for elevation at two different speed steps. Thus, if the hip angle is between 2° and 30° the elevation happens slow, while elevation happens faster when the angle is larger than 30°. This functionality was implemented such that obtaining an ergonomic posture would happen faster when the posture is far from ergonomic. The difference in the two speed steps will be elaborated in section 4.7.1. The flowchart in Figure 4.11 presents how the robot response model is implemented.

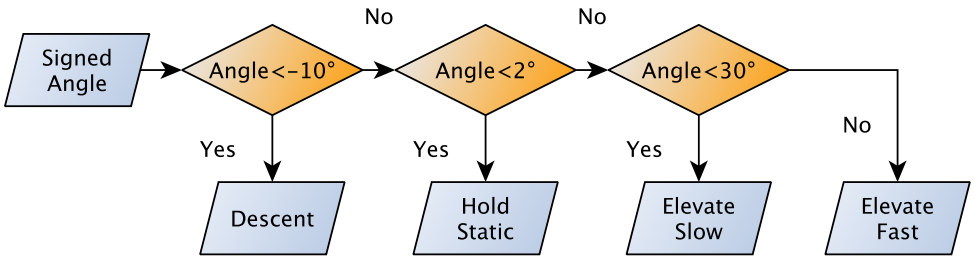


Figure 4.11: The flow of the implemented robot response model in the IS.[13]

4.7.1 The Robot Control Node

The purpose of this node is to have a standard interface between the robot application and the Intera SDK running on the robot. The node subscribes to a "robot command" topic; and based on the commands published to the topic, the robot_control_node calls the right control command from the Intera SDK. Intera SDK is implemented using rospy and, thus, consists of python scripts while the majority of the robot application is implemented in C++. In order to facilitate interfacing between the robot application and Intera SDK, the robot_control_node is implemented in python and subscribes to the ROS topic published by the C++ scripted robot_adaptation_node.

The robot_control_node is implemented as a two part solution consisting of a python module and an executable. The module contains the definitions of all the necessary control functions while the executable is the actual ROS node. The ROS node subscribes to the /ergonomic_adaptation/adaptive_action topic and has a callback function in which a function from the module responsible for initiating the right robotic action is called. The function called takes the message parameters "action" and "speed-step" from the topic as input and calls a designated function corresponding to the action argument. In that way the functionality of the function is a decision tree. The designated action functions then calls a "move to cartesian point" function with a point slightly higher or lower than the current position of the endpoint of the robot and a cartesian move speed dependent on the action. Two possible speed steps have been defined for the elevation action. The maximum linear speed is set to 0.3 $\frac{m}{s}$ for the slow elevation and 0.6 $\frac{m}{s}$ for the fast elevation.

During the implementation of this node some issues were encountered. The first problem encountered was related to the execution of the commands parsed to the robot. The movements

[13]Source: van den Broek and Moeslund (2020)

parsed to the robot are planned using a cartesian interpolation between the current pont and the goal point. However, for some reason this interpolation was not possible with any start configuration of the robot. Therefore, a start configuration that proved useful for overcoming this issue was added as the first step in the node.

Another issue encountered was the fact that the robot is not interrupted when it is executing a movement. This fact combined with queues related to the received messages from the ROS topic and the callback function resulted in an accumulating delay of the execution of the robot control commands. The problem was that during the time the robot is executing a movement several control messages are received and saved for later execution. When the robot has finished the first execution it will move on to executing the next in the queue during which even more control messages are accumulated in the row. This eventually means that the executed movements will be increasingly outdated the longer the robot application runs. The source of the delay was discovered by enumerating the control messages sent and observing that the messages were reported as an unbroken sequence by the robot_control_node despite the delay caused by the execution of the movements.

To fix this problem, several unsuccessful trails were made with queues for the different topics and callback functions set to 1, However, the issue was eventually fixed when the queue size for the subscriber to the /ergonomic_adaptation/adaptive_action topic in the robot_control_node was set to 1. Subsequently, the robot was able to successfully adapt the height of a work surface by monitoring the human operator as shown in Figure 4.12.

Figure 4.12: The end of the robot arm simulating the work surface. The height of the end effector is continuously adapted vertically according to the hip angle as indicated by the red arrows.

The adjustment of the height of the end effector in the current implementation happens stepwise with approximately one second breaks between the adjustments.

4.8 Evaluation of the IS

This section evaluates the implemented IS in respect to observations and considerations made and the initial requirements specified in section 2.2.

When running the system, evaluations of the frame rates were made. The frame rate was evaluated by examining the average rate across 20 frames with which messages are published to the different ROS topics in play in the system. The first topic examined is the /cv_camera/image_raw topic which represents the rate with which the cv_camera_node publishes images. The average rate was 10.970 calculated across 20 samples. The rate of the output of the openpose_ros_node published to the /openpose_ros/human_list topic was 10.965. The rate of publication to the /angle_calculation/body_angles topic was 9.288. Lastly, the publication to the topic /ergonomic_adaptation/adaptive_action also happed with a rate of 9.288. A rate of approximately nine inputs per second should be sufficient for this application as the movements of the robot are supposed to be executed at limited speed. Furthermore, the robot does not receive new inputs while it is executing a command. A small test showed that the execution times for the stepwise adjustments varies dependent on where in the robot workspace it is performed. The recorded execution times varied from 0.4 to 1.0 seconds. This means that the requirement of 2 commands per second is too low to have a new command available for the robot at all times; however, with a rate of approximately 9 commands per second sufficient commands are published.

Similar to the frame rate, the delay of the system response is valuable to know. However, with a modular system as ROS, this measure is hard to obtain. The nearest indicator might be calculating the combination of the average publishing intervals, which gives 0.397 seconds. This is below the set requirement of a less than 1 second delay; however, there are many unknowns that are not included in such a calculation. Also, it should be noted that the delay was clearly perceivable by humans, which suggests that it most likely was larger than the calculated 0.397 seconds. However, the delay in the current system is small enough that the adaptive moves are executed within a few seconds (1-2 seconds) after a change in the posture. While the delay does not necessarily comply with the set requirement, it seems tolerable for this application.

All in all this means that the real time capabilities of the system partially comply with the requirement specified in section 2.2.

One of the requirements specified is that the system should have a non-invasive vision-based human pose estimation system. As described in this chapter, this requirement has been met through the use of the Logitech$^{\text{ő}}$ QuickCam and the OpenPose library. The setup is able to detect the key points of the human pose. However, the OpenPose has some flaws. Every now and then it will recognise things that seemingly resemble a human body and detect false key points on the objects. Also, it sometimes exhibit problems with identifying certain key points like the elbow and knees when the limbs are perfectly straight. This issue presumably arises due to lack of distinctive features for the positioning of e.g. the elbow. Consequently, oscillations of the key points along the limbs can sometimes be observed. Nonetheless, this only seems to be an issue when the limbs are perfectly straight which does not happen frequently for the arms in the use case, and for the legs the observed angle does not change due to this oscillation.

Ergonomic assessment capabilities is also specified as a requirement. Though basic, the IS fulfils this requirement partially by assessing the hip angle; however, the IS does not include a measure for repetitiveness. The ergonomic assessment capabilities should be advanced in the next cycle of the work such that it includes a more elaborate measure for the posture and a measure for repetitiveness.

The next requirement is the adaptive behaviour. The system is able to successfully adapt the height of a work surface dependent on the hip angle. The adaptation approach used was not a direct optimisation of the posture but rather a direct mapping between certain observed conditions and suitable robot movements. This approach resembles the approach used by Shafti et al. (2019) presented in section 1.4.3. This approach is a relative straight forward way to

implement robotic behaviour that guides the human to more ergonomic conditions. However, the method has its limitations. One of the limitations of the method is that it builds on assumptions about how certain robotic movements will affect the posture of the human and relies on these assumptions to be true to reach the optimal positioning of the work surface or the meat piece. Another related limitation is that this method does not provide any means of knowing the exact configuration of the optimal human posture or positioning of the meat independent of the iterative adjustments. This also means that it is hard to include and adapt to a measure for repetitiveness. In order to find the optimal posture, the adaptation has to happen based on optimisation of e.g. a differentiable cost function. In the context of this project this essentially means that a differentiable ergonomic measure is needed since this is the aspect the system strives to optimise according to.

Another aspect of the adaptive behaviour worth addressing is the manner in which the robot carries out the adaptation. As mentioned in section 4.7.1 the robot carries out the adaptive movements realtime in abrupt steps. This stepwise adaptation is partially a consequence of the uninterruptible execution of the movements and partially due to the parameters set for the execution of the movement. Possibly the adaptation of the movements would appear more pleasant and easier to adjust to for the human operator if the movements are executed more smoothly. However, considering the use case at hand, this continuous realtime adaptation may be inconvenient and even unwanted behaviour. If the robot suddenly moves during the butchers processing operations, it will likely complicate the task and possibly introduce errors or unsafe situations for the butcher. Furthermore, there are examples in the literature that indicate that autonomous robot behaviour in Human-Robot Interaction (HRI) increases the cognitive load for the human, especially when the robotic actions are incomprehensible for the human, which to some extend must be true for the unannounced unexpected adaptive moves by the robot in the IS (Zhang et al., 2015, 2017). For these reasons, it is preferable that the robot only adapt its initial positioning of the meat but does not continuously execute adaptive movements. Since the robot then only is able to optimise the position of the meat and, thus, the ergonomics at specific time instances, it is also insufficient to adapt the position of the meat through reactive iterative operations as described above. The system needs to be able to determine an ergonomic position for the meat independently of the current body posture. Thusly, the implementation of ergonomic positioning at given time instances also calls for an optimisation approach as the one discussed above.

As a result of these reflections, the requirements for realtime and ergonomic assessment capabilities and adaptive behaviour are adjusted. The robot should not continuously adapt its movements but should optimise the position at the start of each processing cycle. The ergonomic assessment should still continuously monitor the human operator. As mentioned, the ergonomic measure used for the adaptation of the position of the meat should be differentiable for the purpose of optimisation.

The next requirement to address are the 2D capabilities. The designed system does have 2D capabilities in respect to monitoring the human. The movements are monitored in the sagittal body plane as previously described. The robot's adaptive movements, however, are only carried out in one dimension, namely, along the vertical axis. Though it is common for the ergonomic assessment methods to focus on the movements in the 2D sagittal plane in respect to the posture, there are aspects of the pose considered in ergonomic assessments that will not be included if only the sagittal plane is monitored. Furthermore, if the robot should place the meat in an optimised position in respect to the human in the 3D cartesian space, the 3D position of the human is necessary even if the adaptation only happens in respect to ergonomic measures deducted from 2D spaces. For this reason, and with the purpose of advancing the system, the

requirement of 2D capabilities is revised to a requirement for 3D capabilities. For the IS, it is not possible to evaluate the Euclidean precision of the detected key points as only angles were considered and were calculated based on key points specified in pixel coordinates.

The last requirement specified in this project was the need for a collaborative robot manipulator. This requirement has been met through the use of the Sawyer robot by Rethink Robotics, which has force control, padding and a flexible and interactive design. However, it should be mentioned that this robot would not be suitable for application in the actual use case. This is due to the fact that the robot's payload is too small compared to the weight of the processed belly pieces.

Table 4.1 summarises this evaluation by provides an overview of the requirements and, if relevant, key words for the mentioned revisions. The system as an entity works as intended for the

1	Non-invasive vision system	✓	
2	Ergonomic assessment	✓	More Elaborate Measure
3	Repetition measure	✗	
4	Adaptive behaviour	✓	From task to task, analytical
5	2D capabilities	✓	3D capabilities
6	Precision of ±2.5 cm	✗	
7	Realtime	✓	Only monitoring
8	Command Rate ≥ 2	✓	≥ 3
9	Delay ≤ 1 s	✗	
10	Collaborative Manipulator	✓	

Table 4.1: A checklist of the requirements. The last column provide keywords related to the revision of the requirements.

IS. At its current state the system is able to detect the human pose, evaluate the hip angle as long as the human is oriented with either hip facing directly towards the camera, and move the robot end effector accordingly simulating the work surface with the meat. The work with the system implemented as the IS has been presented at the ACM/IEEE International Conference on Human-Robot Interaction and has been published in (van den Broek and Moeslund, 2020). There are still many potential advancements to investigate. Chapter 5 presents the adjusted design and the advancements made in order to meet some of the reviewed requirements and technical issues addressed in this section.

Chapter 5

Advancing the Ergonomic Adaptation

This chapter describes the second iteration of research work carried out in this project. The focus of this iteration is to advance the IS in respect to a selection of aspects pointed out in section 4.8.

5.1 The Goal of the Advancements

As discussed in section 4.8, a selection of advancements are needed in order to achieve a setup that enables the introduction of a measure for repetitiveness, or more specifically, a method for reducing repetitiveness during Human-Robot Collaboration (HRC). The necessary advancements in order to create a method for ergonomic optimisation of robotic movements that takes repetitiveness into account are related to the parts of the system marked with the red squares in Figure 5.1 and Figure 5.2.

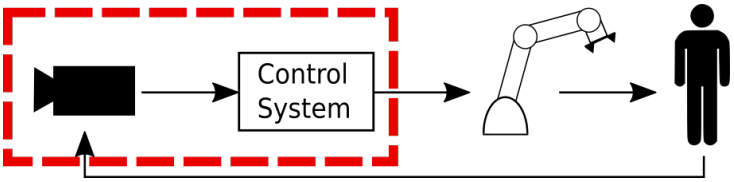


Figure 5.1: The part of the system subject to the advancements.

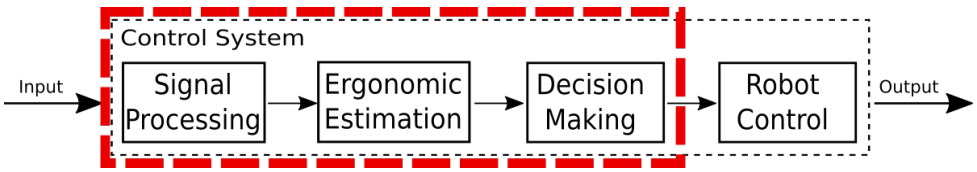


Figure 5.2: The part of the control system that is advanced during this iteration of research work.

The aim of the advancements is to enable the system to successfully place a piece of meat in an ergonomically optimised position in front of the butcher each time a new piece should be processed. The use case is thus brought to a higher level of complexity, where adjustment in 3D and in respect to more ergonomic factors is the goal.

Based on the considerations described in section 4.8 a list of the necessary advancements can be compiled.

1

A more advanced ergonomic assessment method is needed and the method should enable optimisation. Thus, ideally, the ergonomic assessment method should provide an ergonomic measure as a differentiable function of the considered ergonomic factors.

An ergonomic assessment in the described convenient form can be obtained by using the method presented by Busch et al. (2017) to obtain a differentiable version of the Rapid Entire Body Assessment (REBA) method. This ergonomic measure is also based on more postural components. The creation of a differentiable REBA is described in section 5.4 and the utilisation of the method is presented in section 5.5.

2

3D capabilities are needed for obtaining the necessary information for the REBA, but is also necessary for the robot to be able to find the correct placement of e.g. the meat piece in the context of the use case.

In order to obtain the mentioned 3D capabilities a two-fold advancement of the system is needed. First, a sensor that provides 3D data is needed. This will be addressed in section 5.2. Secondly, signal processing that provides 3D pose information of the human is needed. This point is described in section 5.3.

3

The method for adaptive robot movements should include a measure for repetitiveness that ensures generation of movements that facilitate lower repetitiveness for the human in the tasks carried out.

In order to introduce a measure for repetitiveness an investigation of this aspect in relation to the use case is needed as well as an examination of how repetitiveness is evaluated in literature. These points are addressed in section 5.7. Subsequently, a method for introducing repetitiveness limitation in adaptive robot movements is proposed and presented in section 5.8.

The sections describing the advancements of the different elements appear according to their order in the pipeline when going from sensor input to the adaptive robot movements: first the sensor and the signal processing, then the advanced ergonomic assessment method and its usage, and finally the work with introducing a measure for repetitiveness.

5.2 The ZED2 Camera

As a consequence of the revised 3D requirement, a new sensor, which supports 3D data recording is required. The camera chosen is the ZED2 camera from Stereolabs, which is a stereo camera (see Figure 5.3). The ZED2 camera was chosen for its large field of view and depth range which provides flexibility in respect to the positioning of the camera. The camera has variable resolutions and frame rates which are correlated in a trade-off relationship; however, the highest available resolution is 4416×1242 pixels and the highest frame rate possible is 100 Hz. The camera consists of two RGB cameras with a horizontal, vertical and diagonal Field of View (FOV) of 110°, 70° and 120°, respectively. The range for the depth imaging is 0.2-20 meters distance, and the depth imaging is subject to the same FOV as the individual RGB cameras. Likewise, the available resolutions and frame rates for the depth sensing are equal to those for the native video. Besides the imaging sensors, the camera also contains motion and position sensors such as, e.g., accelerometer and barometer. The camera connects through a USB 3.0.

Figure 5.3: The ZED2 stereocamera from Stereolabs used to obtain 3D data.[1]

The ZED camera is a passive stereo camera. This means that it through the use of two RGB cameras placed adjacent is able to detect the distance from the camera to points in the recorded scene. The ZED2 camera uses neural depth sensing, which means that synchronised images from the two RGB cameras are used in neural networks in order to obtain the depth map of the scene.

For more information about the specifications of the ZED2 camera please see (Stereolabs, 2020).

5.2.1 Setup for the ZED2

The ZED2 camera comes with a ZED SDK[2] which provides both basic and more advanced functionalities. In order to use the SDK the existing Docker image has been edited to accommodate dependencies of the SDK and include the SDK. In this process, the CUDA tool kit version has been changed to 10.2 and OpenGL, libcuvid and libpcl have been included to support the use of point clouds and visualisation. During this work the ZED SDK 3.1 for CUDA 10.2 and Ubuntu 16.04 is used.

To be able to run a container and use the ZED camera inside the Docker container, the Docker engine, Docker API and the Nvidia Container Toolkit have to be updated, and the USB busses and virtual video device nodes have to be parsed to the container at launch.

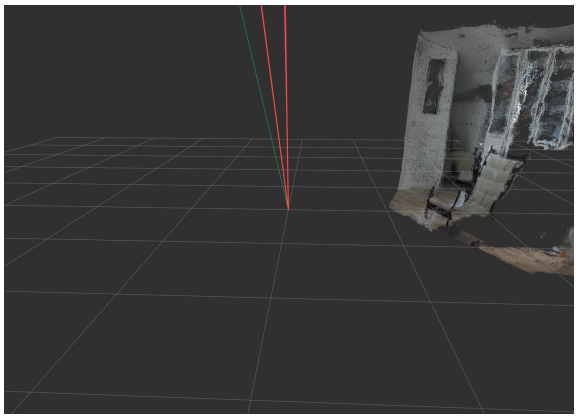
For convenient interfacing in Robot Operating System (ROS), the zed-ros-wrapper[3] provided by Stereolabs is also used and has been added to the Docker image. The wrapper contains a large range of parameters which provide an easy way of specifying the wanted functionalities. As default the wrapper published images in the image topics with a 720HD resolution and with a frame rate of 15.0 Hz. The sampling rate is 30.0 Hz.

5.3 3D OpenPose

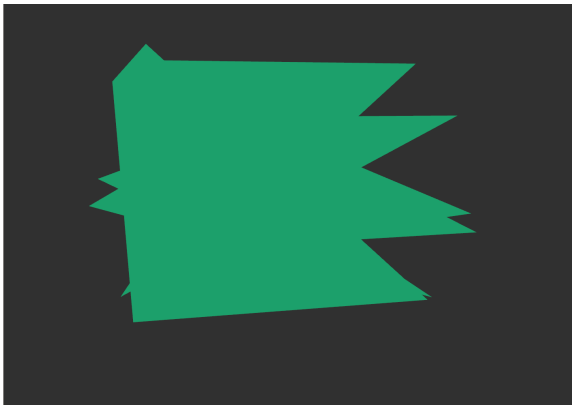
In order to obtain the human pose key points in 3D, a ROS wrapper for OpenPose that supports 3D point detection is needed. For that reason, the available selection of ROS wrappers for OpenPose has been reviewed in order to identify which wrappers fulfil this requirement while being suitable for working together with the ZED2 camera. A total of five wrappers have been investigated of which only two provide 3D capabilities, and one of the two is deprecated. However, the remaining ros_openpose wrapper created by Ravi Prakash Joshi[4] unfortunately did not directly support the ZED2 camera. Consequently, work with adapting the wrapper and creating the necessary files for interfacing with the ZED2 camera has been done. To ensure

[1] Source: Stereolabs, https://cdn.stereolabs.com/assets/images/zed-2/zed-2-front.jpg
[2] https://www.stereolabs.com/developers/release/
[3] https://github.com/stereolabs/zed-ros-wrapper
[4] https://github.com/ravijo/ros_openpose

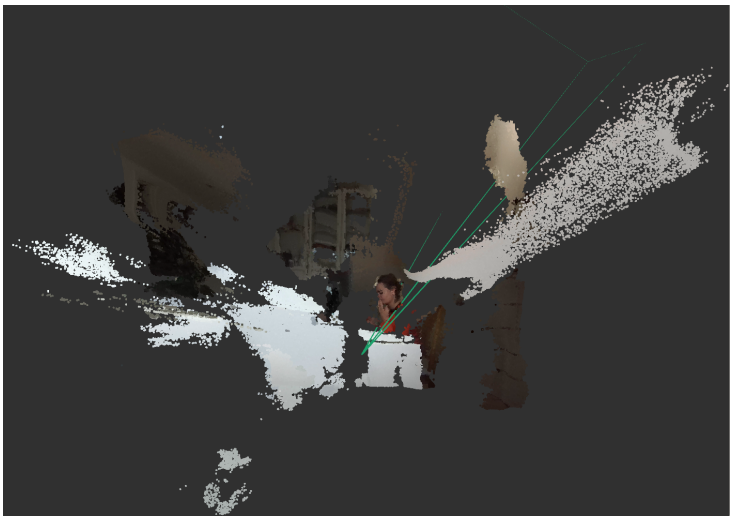
interfacing with the ZED2, a new ROS launch file including ZED2 specific arguments was created. The arguments provided include the name of the RGB image topic, the name of the depth map image topic, a camera_info topic containing camera parameters and the transform frame specifying the frame in which the data is captured by the camera. The wrapper also contains files for setting up visualisation in RViz which, likewise, was created for the ZED2 camera with the necessary transform frame and the point cloud topic of the ZED2 camera specified.



(a) An off-set and faulty projection of the human model.



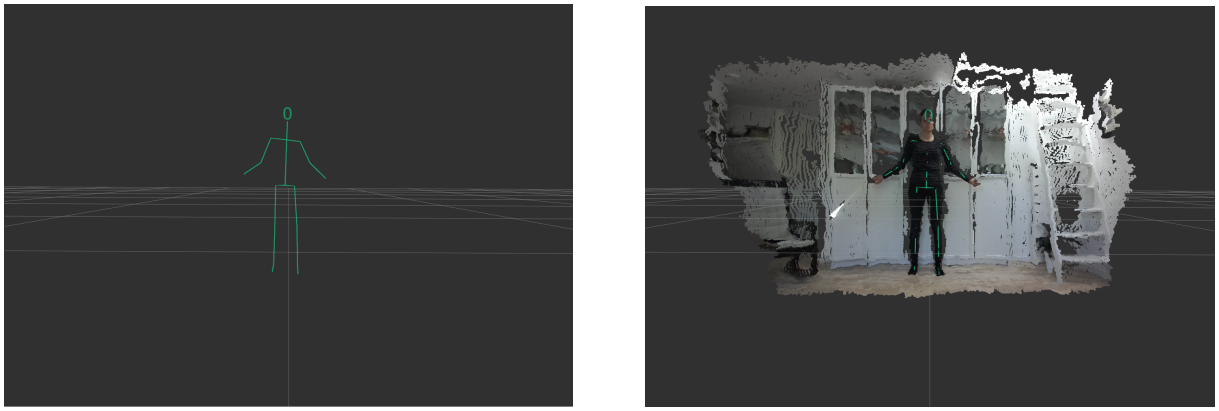
(b) All body segments are seemingly visualised in one point.



(c) The visualised human model seems out of proportion and projected out from an anchor point.

Figure 5.4: Issues visible in the Rvis visualisation of the output of the ros_openpose wrapper.

With the created files the wrapper was able to run with the ZED2 camera, however, from visualisation (see Figure 5.4) and inspection of the returned key points it was evident that the key point detection was faulty. The problem turned out to be that the depth map image published by the ZED ROS wrapper was in the wrong format. By changing the openni_depth_mode parameter in a file in the ZED wrapper the depth map format was changed from 32-bit float in meters to 16-bit uchar in millimetres which was needed for the ros_openpose wrapper. Figure 5.5 shows visualisation in RViz after successfully interfacing with the ZED wrapper. The adapted files for the ZED2 camera and the knowledge about how to make the wrapper run with the camera have been shared with Ravi Joshi, and the ZED2 support has now been added and made available to others on the public git repository.



(a) The detected 3D human pose model.

(b) Visualisation of the detected human model aligned with the point cloud detected by the ZED SDK.

Figure 5.5: Successful visualisation of 3D human body model.

The way the ros_openpose wrapper determines the 3D key points is by projecting the 2D key points found by OpenPose onto the depth map provided by the ZED wrapper. In practice, the intrinsic camera parameters, the focal lengths and the principal point, and the x and y pixel coordinates of the key points are used to to calculate the relation between the distance from the camera and the x and y-coordinates, respectively, in the in the camera frame. These relation values are then multiplied with the value of the depth map at the pixel x and y-coordinates to obtain the mentioned x and y coordinates in 3D of the key points. The z-coordinate is given directly by the value of the pixel of the depth map at the x and y pixel coordinates. Distortion is not considered in the calculations in the ros_openpose wrapper.

5.4 Constructing the Differentiable REBA Method

This section describes how a differentiable REBA useful for optimisation was constructed. The differentiable REBA is constructed based on the original REBA method presented by Hignett and McAtamney (2000) (see Figure 5.6) following the approach presented by Busch et al. (2017). The original REBA as shown in Figure 5.6 is a discrete function based on intervals and table lookups. The function in Equation 5.1 describes the individual score for the trunk angle. Clearly, it is a discrete function, and the scores for the remaining individual joints considered in REBA are given by similar discrete functions.

$$C_{\text{trunk}}(q_{\text{trunk}}) = \begin{cases} 3 & q_{\text{trunk}} < -20 \\ 2 & q_{\text{trunk}} \in [-20, 0) \\ 1 & q_{\text{trunk}} = 0 \\ 2 & q_{\text{trunk}} \in (0, 20] \\ 3 & q_{\text{trunk}} \in (20, 60] \\ 4 & q_{\text{trunk}} > 60 \end{cases} \tag{5.1}$$

Where q_{trunk} is the trunk angle, and C_{trunk} is the individual REBA score for the trunk.

[5]Source: Neese Consulting, https://www.physio-pedia.com/images/a/a6/REBA.png

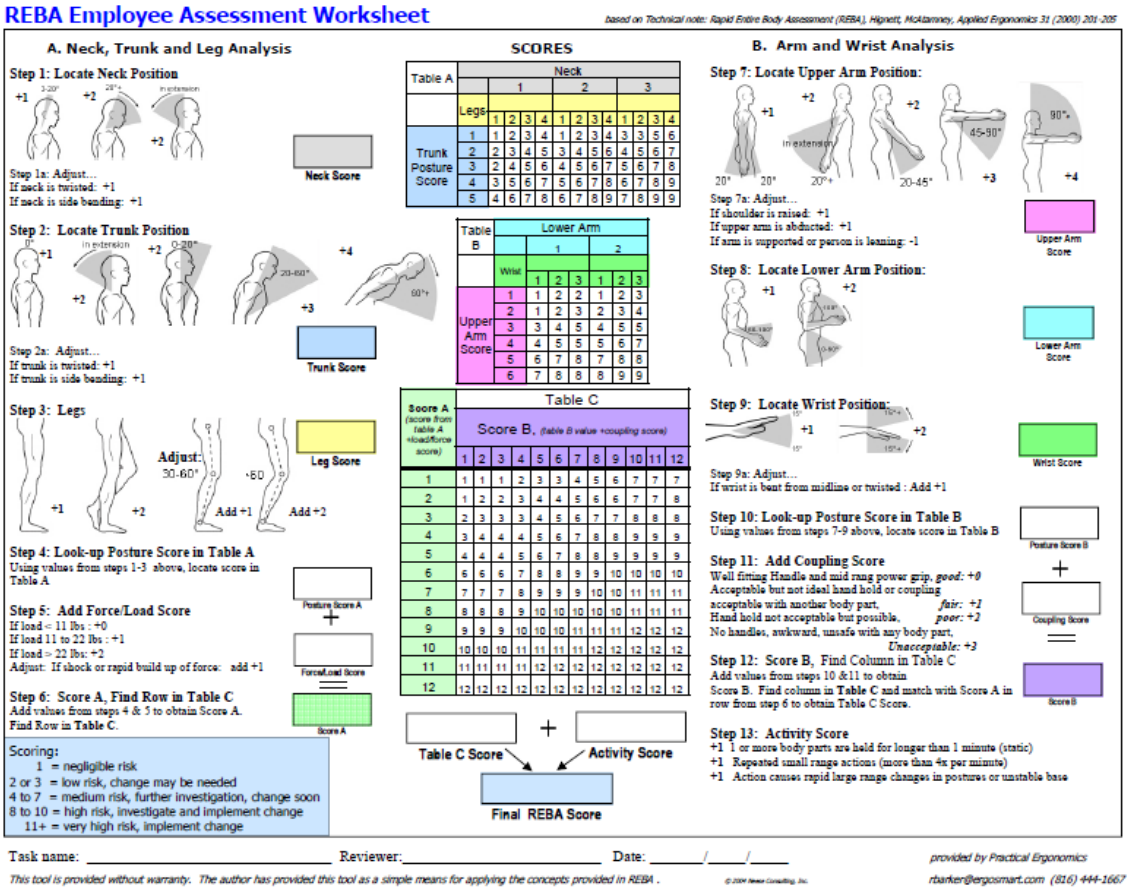


Figure 5.6: The REBA method presented as a worksheet.[5]

As mentioned earlier, such discrete functions are inconvenient for optimisation and robot control. A continuous differentiable function is preferable. Busch et al. (2017) have presented a method for obtaining a differentiable function for the REBA score by fitting a differentiable function to the original discrete function. Though their approach for creating a differentiable REBA method is described in their article, their method is not described thoroughly. There are several unknowns in regard to their choices made with respect to constructing the REBA. When known and applicable for the context of this project, the choices made by Busch et al. (2017) are used. When, however, insufficient details are available in the article, choices are made and arguments are provided.

The general approach described by Busch et al. (2017) is to fit a sum of weighted polynomials as described by Equation 5.2 to the REBA method.

$$C_{\text{posture}}(q) = \sum_{i=1}^{n} w_i Q_i(q_i) \tag{5.2}$$

Where n is the number of joints considered in REBA, $Q_i(q)$ is a second degree polynomial of the angle q_i at joint i, and w_i is the weight of $Q_i(q)$ and is a scalar. The discrete function for the individual joint scores $C_i(q_i)$ is thus represented by a differentiable polynomial $Q_i(q_i)$.

Busch et al. (2017) approach is as follows: First, second degree polynomials are fitted to the scoring levels described for each of the monitored angles in the REBA. Subsequently, a selection of postures with corresponding REBA scores spread across the one to 12 score interval are used to identify the weights for the individual polynomials. Following the same approach a second

degree polynomial was fitted to the scoring levels for each body part included in the REBA. The first strategical choice which is not addressed by Busch et al. (2017) is how the data used for the polynomial regression were based on the information in the original REBA. In the method a score is given for an interval resulting in the discrete discontinuous plateaus; but when using the information for the regression, the provided score has to be assigned to a specific angle. Therefore, a policy for how this is done is needed in order to ensure that the same approach is used for the creation of the basis for each polynomial regression. As the information approach used in Busch et. al's work was not disclosed, the following rules were set and followed:

1. If a score level is provided for one specific angle, this is used as a point.
2. If a score level is provided for a closed interval, the score value is assigned to the angle deviating the most from the neutral position.
3. If a score level is given for an open interval, the score is assigned to the angle at the maximally deviating angle from the neighbouring lower interval + $\frac{1}{2}$ of the previous interval size.
4. If a score level is provided for an open interval with no neighbouring lower interval, the score is assigned to the nearest angle with a specified score + 20°

The scoring levels always seem to be symmetric around a given angle with the minimal score except for the fact that they might not be specified to even deviation angles on both sides of the given angle (Figure 5.7 exemplifies this). The choice was made to create symmetric data points based on the indicated tendency to use as the base for the polynomial regression. Consequently, points found for open intervals might be neglected and substituted by a point symmetric to a point specified for a closed interval on the opposite side of the given minimum angle.

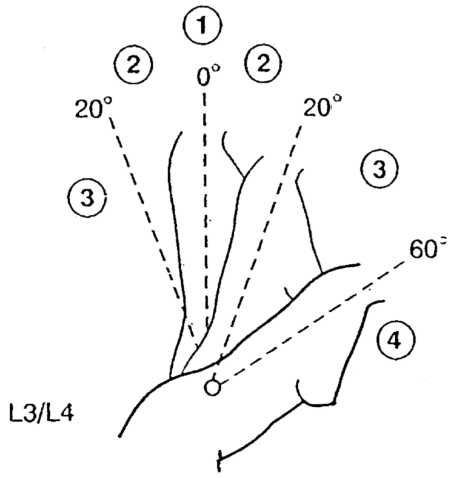


Figure 5.7: The score levels for the trunk in the original REBA. Note the symmetric tendency around the angle of 0° which is discontinued on one side.[6]

Following the rules described, data was generated and polynomial regressions were done, and second degree polynomials were obtained. Figure 5.8 shows an example of a polynomial regression. The plot shows the discrete score levels of the original REBA as well as the fitted differentiable score function for the trunk angle. From the plot it is clear how the differentiable second degree polynomial resembles the original discrete function. The found polynomial coefficients and the R^2 for the second degree polynomial for the different body parts are provided

[6]Source: (Hignett and McAtamney, 2000)

in Table 5.1. The data points and the plotting of all the found second degree polynomials can be found in Appendix C.

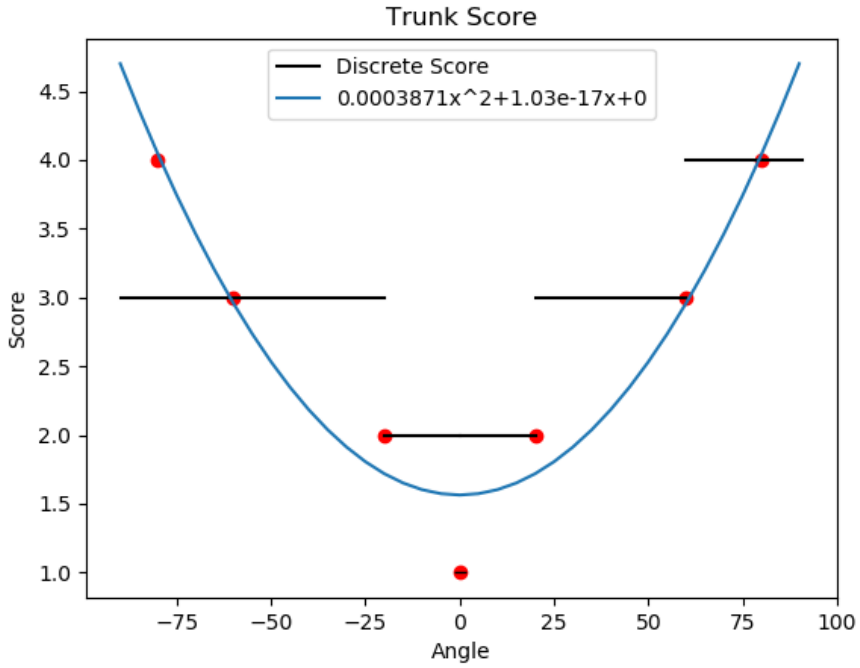


Figure 5.8: The fitting of a second degree polynomial to the discrete REBA score levels for the trunk angle. The red dots mark the generated datapoints used for the polynomial regression.

Score	**a**	**b**	**c**	R^2
Trunk	3.87080868e-04	1.03037038e-17	0.00000000e+00	0.935
Neck	0.0025	-0.05	0.0	1.000
Legs	4.76190476e-04	-1.14290758e-18	0.00000000e+00	0.918
Upper Arm	2.33132981e-04	-3.37406021e-18	0.00000000e+00	0.960
Lower Arm	0.000625	-0.1	0.0	1.000
Wrist	1.48148148e-03	1.38998410e-18	0.00000000e+00	1.000
Side Flex	2.50000000e-03	5.42697202e-18	0.00000000e+00	1.000

Table 5.1: The polynomial coefficients and the R^2 of the fitted second degree polynomials for the scores for the different body parts. The letters refer to the coefficients in a polynomial on the form $f(x) = ax^2 + bx + c$.

Besides the scoring based on the angles of bodyparts, the posture scores in the REBA also includes a few other scores for twists, side flexions, abduction, raised shoulders, arm support, gravity assisted upper limb posture, and one leg support. The twists in body parts and the raised shoulders are rather difficult to detect with the used method; also, it is not stated in the article that Busch et al. (2017) included this in their method. Due to this, the twist measures for all body parts are ignored. The score for the side flex in the torso and head, the abduction of the arm and the deviation of the wrist are all represented by the second degree polynomial called side flex in Table 5.1 and Appendix C, which will be included in the final REBA one time for each of the four scores. The arm support, gravity assisted upper limb posture and one leg support can all be neglected based on the context of this work. The overall posture in which the butchers are working is standing more or less upright, with both feet on the ground. As a result, the last three mentioned scores can be ignored.

The REBA also includes a few scores that are not related to posture. These scores are coupling (grasp), force/load score and activity score. In the context of meat being presented to the butcher by a robot, the only grasp the butcher has to do is the cutting tool and handling of smaller parts. These would both be scored with 0 and therefore this can be neglected.

The force/load score requires a bit more intricate considerations. The score is determined based on handled weight. The smaller pieces that are cut off would not likely exceed the 5 kg load which is the upper limit for a score of 0. Thus, the handling of the smaller pieces will not cause an increase in the score. Next the forces used for cutting would be relevant to consider. The force could be represented by determining the weight that affected by gravity results in the same forces as the force exerted to cut the meat. Sources from literature are used to get information about the realistic cutting forces. The forces needed for cutting the meat most likely depends on several factors. It is possible that the temperature of the meat is one of these factors. According to the Regulation(EC) No. 853/2004 of the Europa Parliament and of the Council - *laying down specific hygiene rules for food of animal origin* annex III chapter V the meat has to have a temperature of 7 °C during cutting and boning, and the room temperature during handling has to be 12 °C(Council, 2004). In the literature research can be found that investigates the cutting forces in meat at different temperatures (Brown et al., 2005; Hägg et al., 2012). However, the literature is not unanimous. The forces differ in the range 10-560 N in similar temperature ranges for different setups indicating that the setup has a large impact on the measured forces. With the differences in forces being so large and none of the setup closely resembling the setup at the slaughterhouse in the context of this project, it makes it impossible to determine a realistic force exertion. In a final implementation one might have to investigate the exerted forces for the use case at hand and deduct a matching score, but in this project the force will be ignored due to insufficient knowledge. Besides, the load/force score would only be included as an added constant scalar if the forces are assumed to be constant due to the lack of realtime measuring. This means that while it might affect the size of the score it will not affect the global minimum of the function, which is the property exploited when it is used for optimisation.

The last score is the activity score. In our use case the activity score will be solely dependent on the frequency criterion which is constant. This means that this also would be an added constant scalar and would not influence the location of the minimum of the function.

After these considerations, the final sum of weighted polynomials consists of ten weighted polynomials for the trunk score, trunk side flexion score, neck score, neck side flexion score, legs score, upper arm score, upper arm abduction score, lower arm score, wrist score, wrist deviation score, respectively.

In order to find the weights of the polynomials a selection of poses at different REBA score levels are created and used in a multiple linear regression. The highest score obtainable in the original REBA method using the parameters included in the differentiable REBA is 11; therefore, 22 poses, two at each score level in the interval of 1-11 were created. The poses can be found in Appendix D. Table 5.2 contains the determined weights as well as the R^2. The found model represents the tendency in the data points rather well as can be deducted from the value of R^2.

w_1	w_2	w_3	w_4	w_5	w_6	w_7	w_8	w_9	w_{10}	R^2
0.787	0.737	1.094	1.264	0.746	0.645	0.561	0.371	0.287	0.932	0.961

Table 5.2: The weights determined for the sum of weighted polynomials and the coefficient of determination of the fitted multiple linear regression.

The created differentiable REBA might not actually result in the exact same score for different assessed postures as the original REBA; however, the important property is that it will have the minimum at the same location as the original REBA.

5.5 Optimising the Posture

The differentiable REBA is optimised with respect to angles. One may observe that the REBA score as a function of the angles is a convex function. This is known as it is a sum of positively weighted polynomials with second order coefficients, $a > 0$. The formal definition of convexity for a function is:

A function $f : \mathbb{R}^d \to \mathbb{R}$ is convex if $\forall\ a, b \in \mathbb{R}^d$ and $\forall\ 0 < t < 1$,

$$f(t \cdot a + (1-t)b) \leq t \cdot f(a) + (1-t)f(b) \tag{5.3}$$

For functions of a single variable such as the second order polynomials for the scores for the individual body parts in the differentiable REBA, the convexity can be determined by examining the second derivative of the function. This is given by the rule that a function $f : \mathbb{R} \to \mathbb{R}$ is convex if and only if its second derivative $f''(x) \geq 0$ everywhere. For the found individual weighted second degree polynomials this holds as the second derivative of a second order polynomial with a positive quadratic term is a positive scalar; and as this is multiplied with a positive weight, it remains positive. Having established the convexity of the individual weighted polynomials we can conclude that the differentiable REBA, the sum of weighted polynomials, is in fact a convex function given the knowledge that the sum of convex functions will also be convex.

The convex optimisation problem to be solved is given by Equation 5.4.

$$\min_{q} \quad \sum_{i=1}^{n} w_i Q_i(q_i) \tag{5.4}$$

Where n is the number of joints considered in REBA, $Q_i(q)$ is a second degree polynomial of the angle q_i at joint i and w_i is the weight of $Q_i(q)$ and is a scalar.

The optimisation is done using a quadratic cone programming function from the CVXOPT package for python. Consequently, the differentiable REBA, the sum of weighted polynomials, is represented in the form:

$$\frac{1}{2}x^T P x + q^T x \tag{5.5}$$

The matrix P is constructed as a diagonal matrix with entries calculated from the quadratic coefficients (a) and the weights (w) in the following way: $P_{\text{nn}} = a_n \cdot w_n \cdot 2$. Likewise q is constructed by multiplying the linear coefficients (b) with the weights (w), $q = w \cdot b^T$.

The matrices P and q are parsed to the optimiser as an unconstraint optimisation problem. The identified optimum will give the optimal body angles for minimising the differentiable REBA. The identified optimal angles are listed in Table 5.3, and the resulting optimal posture can be seen in Figure 5.9.

Trunk θ_t	**Trunk Side** θ_{ts}	**Neck** θ_n	**Neck Side** θ_{ns}
0.0	0.0	10.0	0.0
Right Knee θ_{rk}	**Left Knee** θ_{lk}	**Right Upper Arm** θ_{rua}	**Right Arm Abduction** θ_{raa}
0.0	0.0	0.0	0.0
Left Upper Arm θ_{lua}	**Left Arm Abduction** θ_{laa}	**Right Lower Arm** θ_{rla}	**Left Lower Arm** θ_{lla}
0.0	0.0	80.0	80.0
Right Wrist θ_{rw}	**Right Wrist Deviation** θ_{rwd}	**Left Wrist** θ_{lw}	**Left Wrist Deviation** θ_{lwd}
0.0	0.0	0.0	0.0

Table 5.3: The optimal body angles identified through unconstraint optimisation of the differentiable REBA. The angles are stated in degrees.

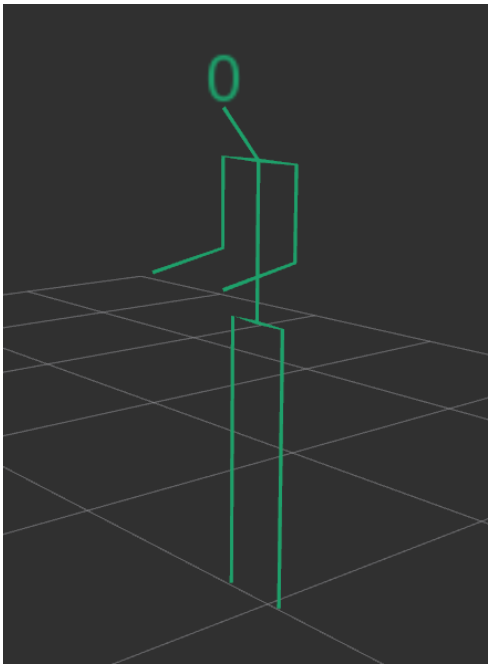


Figure 5.9: The human body model in RViz showing the optimal posture. The uppermost segment is the line from the top centre point of the manubrium to the nose.

By conducting a traditional REBA with the original discrete method of the found optimal posture, it can be confirmed that this posture indeed is an optimal posture that results in the lowest possible REBA score, namely, a score of 1.

In relation to the use case addressed in this project, it is relevant to determine the correct position of a meat piece presented by the robot to the butcher given the found angles. For this purpose forward kinematics of the human body is useful. The method used for determining the position where the robot should place the meat piece is explained in section 5.6.

5.6 Optimal Robot Position

In order to simplify the problem of finding the optimal position for the meat piece, some assumptions are made. Firstly, the optimal position is found as the optimal position for static contact rather than a dynamic interaction. Secondly, the point of contact for the human is assumed to be a point central on the most distal edge of the palm of the hand. This assumption is made as the butchers interact with the meat using their hands and tools. Though the real contact point might be off-set from the point at the palm due to a tool, it is a point realistically close to where the real contact point would be and will serve as such. Another assumption made is that the butcher is right handed.

In order to find the optimal position of the meat, the position of the contact point at the palm has to be identified in 3D Cartesian space given the identified angles. As stated earlier, this can be done with forward kinematics of the human body. Specifically, the kinematics from the key point at the right ankle to the contact point at the distal edge of the right palm. The forward kinematic of the body can be made based on the BODY_25 pose model; however, it is necessary to know the length of all the body segments between the ankle and the contact point. In order to get somewhat realistic measures for how long the segment lengths are perceived to be by the system and to have a system that potentially could obtain this information from an observed subject automatically, a script was coded that calculates the segment lengths from key points detected by the ros_openpose wrapper. Subsequently, the average across ten poses was made to minimise the effect of small errors. The calculated segment lengths from a subject are given in Table 5.4, and these lengths will be used in the onward calculations.

Neck	**Spine**	**Shoulder**
l_n	l_{sp}	l_{sh}
0.196	0.469	0.159
Upper Arm	**Lower Arm**	**Palm**
l_{ua}	l_{la}	l_{pa}
0.271	0.257	0.088
Pelvis	**Upper Leg**	**Lower Leg**
l_{pe}	l_{ul}	l_{ll}
0.111	0.429	0.418

Table 5.4: The lengths of the body segments found from the body key points. The lengths are stated in meters.

Furthermore, it is necessary to know the angles that play a role in the joints between the ankle and the contact point. While the REBA method considers a range of angles in the body, it does not consider all angles; and, thus, by optimising the differentiable REBA, not all the needed angles are identified. Due to this, the assumption is made that all angles excluded in the REBA at all times are $0*^\circ$. This is an acceptable assumption for all angles but one, namely the angle at the ankle. The REBA does include the knee angle, and it is a known fact that bending in the knees when standing forces a change of the angle at the ankle in order to keep the balance (Bonnet et al., 2011). The most important effect the knee angle has on the position of the rest of the body is a vertical translation. However, if the ankle angle is unknown, the resulting translations can only be known if the side opposite to the knee angle in the triangle formed by the ankle, knee and hip key points is assumed to be perfectly vertical. Therefore, the assumption is made that the hip key point is located perfectly vertically above the ankle key point given any knee angle (see Figure 5.10). Though this might not fully portray the complex body movements that happen in order to keep balance when bending the knees, this is a convenient and necessary simplification.

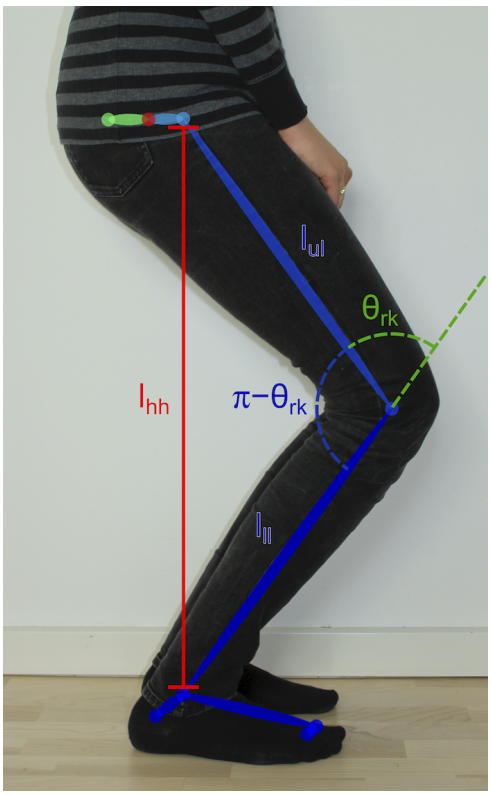


Figure 5.10: The hip height, l_{hh} as a perfect vertical line in the triangle formed by the ankle, knee and hip key points. The upper and lower leg segments, l_{ul} and l_{ll} are marked as well as relevant angles around the knee including the right knee angle included in the differentiable REBA θ_{rk}. The relations depicted in this figure forms the basis for the calculation of l_{hh}.

Utilising the stated assumption the variate height can be found as a function of the knee angle by using the law of cosines.

$$c^2 = a^2 + b^2 - 2 \cdot a \cdot b \cdot \cos(\theta_C) \tag{5.6}$$

But as the effect of the angle is inverse in the sense that the hip height $l_{\text{hh}} = l_{\text{ul}} + l_{\text{ll}}$ for $\theta_{\text{rk}} = 0$, $\cos(\theta_C)$ in this case is set to be $\cos(\pi - \theta_{\text{rk}})$ (see Figure 5.10). Thus, the resulting height of the hip given the length of the upper and lower leg, l_{ul} and l_{ll}, and the knee angle, θ_{rk}, is given by:

$$l_{\text{hh}} = \sqrt{l_{\text{ul}}^2 + l_{\text{ll}}^2 - 2 \cdot l_{\text{ul}} \cdot l_{\text{ll}} \cdot \cos(\pi - \theta_{\text{rk}})} \tag{5.7}$$

Having identified all the necessary segment lengths and the hip height, the forward kinematics can be created by identifying the Denavit-Hartenberg parameters and, subsequently, constructing the necessary transformation matrices for the human body. For the identification of the Denavit-Hartenberg parameters, frames were affixed to the links following the well known convention and orienting the z-axis along the rotational axis and the x-axis along the common perpendicular of the rotational axes of the current and the next following joint. The affixed frames for the considered part of the human body can be seen in Figure 5.11.

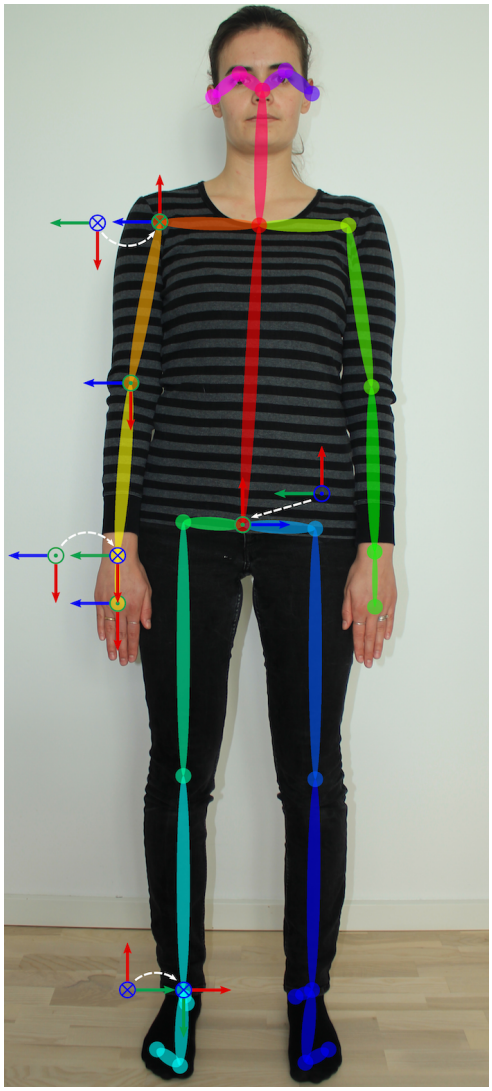


Figure 5.11: The link frames affixed to the human body. An arrow indicates the direction of an axis, a cirkle with a dot in the center indicates an axis pointing towards the viewer while a circle with a cross in the centre indicates an axis pointing away from the viewer. The colours of the markings indicate the axis, where the x, y and z-axis are marked with red, green and blue, respectively. Where two frames are affixed in the same point one is drawn with an offset and a white arrow indicated the correct affixing point.

The Denavit-Hartenberg parameters derived from the affixed link frames can be seen in Table 5.5.

i	α_{i-1}	a_{i-1}	d_i	θ_i
1	0	0	0	$-\frac{\pi}{2}$
2	$-\frac{\pi}{2}$	$\sqrt{{l_{ul}}^2 + {l_{ll}}^2 - 2 \cdot l_{ul} \cdot l_{ll} \cdot \cos(\pi - \theta_{rk})}$	l_{pe}	θ_t
3	$-\frac{\pi}{2}$	0	0	θ_{ts}
4	$-\frac{\pi}{2}$	l_{sp}	l_{sh}	θ_{rua}
5	$-\frac{\pi}{2}$	0	0	$\pi + \theta_{raa}$
6	$-\frac{\pi}{2}$	l_{ua}	0	θ_{rla}
7	$\frac{\pi}{2}$	l_{la}	0	θ_{rwd}
8	$-\frac{\pi}{2}$	0	0	θ_{rw}
9	0	l_{pa}	0	0

Table 5.5: The Denavit-Hartenberg parameters for the KUKA KR6 R700 sixx that is used for the prototype. The distances a_{i-1} and d_i are given in mm. The angles α_{i-1} and θ_i are given in radians.

For each set of Denavit-Hartenberg parameters a transformation matrix of the following form is constructed.

$$T_i^{i-1} = \begin{bmatrix} c\theta_i & -s\theta_i & 0 & a_{\mathrm{i-1}} \\ s\theta_i c\alpha_{\mathrm{i-1}} & c\theta_i c\alpha_{\mathrm{i-1}} & -s\alpha_{\mathrm{i-1}} & -s\alpha_{\mathrm{i-1}} d_i \\ s\theta_i s\alpha_{\mathrm{i-1}} & c\theta_i s\alpha_{\mathrm{i-1}} & c\alpha_{\mathrm{i-1}} & c\alpha_{\mathrm{i-1}} d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.8}$$

The resulting matrices are multiplied as specified by Equation 5.9.

$$T^{Ankle}_{\text{Contact point}} = T_1^{Ankle} T_2^1 T_3^2 T_4^3 T_5^4 T_6^5 T_7^6 T_8^7 T^8_{\text{Contact point}} \tag{5.9}$$

This gives the full transformation from the ankle to the contact point at the most distal edge of the right hand palm in a coordinate system with the same orientation as the camera coordinate system when the human is facing the camera. In this case both coordinate systems will have the x-axis pointing left for the human, the y-axis pointing down and the z-axis pointing backwards. The entries $e_{1,4}$, $e_{2,4}$ and $e_{3,4}$ in $T^{Ankle}_{\text{Contact point}}$ will give respectively the x, y and z coordinates of the contact point in respect to the coordinate system at the ankle dependent on the body angles. The kinematic expressions for the three coordinates of the contact point are provided in Appendix E. For simplicity reasons, only the position of the contact point and not the orientation is considered. For now the meat pieces placed by the robot would maintain the same orientation.

The optimal position of the meat piece in respect to the right ankle point can then be found using the obtained kinematic equations for the coordinates and inserting the body segment lengths and the optimised angles. Doing this the optimal position is found to be $[x = -0.040, y = -0.999, z = -0.345]$.

To get the contact point in camera coordinates, a translational matrix constructed from the detected key point at the right ankle in the camera coordinates should be included. The matrix would depend on the detected point coordinates of the right ankle key point and would be on the form specified in Equation 5.10 given the assumption that the human's coronal plane (see Figure 2.1) is parallel to the image plane.

$$T^{Camera}_{\text{ankle}} = \begin{bmatrix} 1 & 0 & 0 & x_{\text{ankle}} \\ 0 & 1 & 0 & y_{\text{ankle}} \\ 0 & 0 & 1 & z_{\text{ankle}} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.10}$$

With the optimisation established and the forward kinematics for finding the optimal contact point in place, the next step is to investigate how to introduce some measure for repetitiveness into the method.

5.7 Introducing a Measure for Repetitiveness

Repetitiveness, is a well known ergonomic risk that also is included in some ergonomic assessment methods. The relationship between e.g. Musculoskeletal Disorders (MSDs) in the upper limbs and repetition is supported by scientific evidence (Bonfiglioli et al., 2007; Ketola et al., 2001). Minimising repetitiveness is thus desirable.

The word "receptiveness" is used to describe to which extend something is repetitive, or characterised by repetition. When looking up the dictionary entry for repetition, it is describes as the act of doing something again, or repeating an action. If something is described as highly repetitive, it is thus highly characterised by repetition, and it is insinuated that a certain action

is repeated at a high frequency. As mentioned, repetition describes the act of doing something again. This also insinuates that the actions carried out each time are considered alike. In ergonomics definitions of what is considered repetitive are also given. An example is the following statement cited from (Books, 2002).

> *Work is repetitive when it requires the same muscle groups to be used over and over again during the working day or when it requires frequent movements to be performed for prolonged periods.*

In the field of ergonomic assessments the importances of both of these aspects (frequency and variation) are discussed. The first aspect, the frequency, is considered an important risk factor. There are several examples of observational assessment methods that include a frequency factor. David (2005) provides a nice overview of which factors a selection of methods include. In Table 5.6 the result of an investigation into how methods reported by David (2005) to include frequency do so.

Method	**Reference**	**Inclusion of repetitiveness**
The Strain Index (SI)	(Moore and Garg, 1995)	Different intervals of exertions pr. minute map to different ratings, which through table look-ups result in different multipliers used in a multiplication to find the final SI.
RULA	(McAtamney and Corlett, 1993)	Addition of a scalar if repetition higher than 4 pr. min.
REBA	(Hignett and McAtamney, 2000)	Addition of a scalar if repetition higher than 4 pr. min.
NIOSH Lifting Equation	(Waters et al., 1993)	Frequency multiplier is found based on frequency and duration in a look-up table. Multiplier is then used to calculate recommended weight limit and lifting index.
OCRA	(Occhipinti, 1998)	An index is obtained by dividing the
QEC	(Li and Buckle, 2000)	Qualitative assessment
Manual Handling Guidance	(Books, 1998)	Qualitative assessment
FIOH Risk Factor Checklist	(Ketola et al., 2001)	Semi-quantitative method, only considers repetition for the hand. If repetition is present a scalar of 1 is added to a sum of
ACGIH TLVs		Source was not found
Upper Limb Disorder Guidance HSE60	(Books, 2002)	Qualitative assessment
MAC	(Monnington et al., 2003)	Results in a score based on combined assessment with load. Score is added to total score.

Table 5.6: The inclusion method for frequency/repetition in a range of assessment methods.

As Table 5.6 reports, there are mainly three ways frequency is included in the investigated assessment methods. In the qualitative methods it is simply reported as one entry in an assessment summary, and thus not combined with other factors. For the methods that combine frequency

with other factors to obtain a single ergonomic measure, the frequency is either added as a scalar or a multiplication factor. In some cases the magnitude of the factor varies dependent on the recorded frequency.

It could be possible to adjust the behaviour of the robot to lower the frequency; however, in the context of the use case at the slaughter house, this is not realistic. As mentioned in section 1.1, the slaughter houses are under pressure and need to keep a high processing rate.

The other aspect of repetition, variation, is also important to the assessment of exposure to ergonomic risk (David, 2005). As indicated by the quote above sufficient variation might lover the repetitiveness of the tasks carried out. Furthermore, it is well known that variation is healthy for the body and increases comfort (Vink et al., 2009). This is also why many assessment methods punish static postures, which is also the case for the REBA. Variation, however, is usually not directly included in traditional observational methods, where an observer assesses the work situation. This is probably because it is hard to estimate the amount of variation with the bare eye. Nonetheless, there are examples of more advanced video-based observational techniques and direct measuring methods that enable relatively precise posture estimation and thus assessment of posture variation. Examples of included measures for assessing posture variation are distance of movement, angular changes, and velocity and acceleration (David, 2005).

The distance of movement and angular changes are measures that might be useful for introducing variation in ergonomic adaptation of the robotic movements. As mentioned in section 4.8, the suitable approach for the robot adaptation in the context of the use case is to find an optimal placement of the meat for each processing task carried out by the butcher and hold the meat steady at that position for the duration of the processing task. Variation can then for example be introduced by ensuring that the body angles when reaching the meat vary sufficiently from processing task to processing task. At the same time, it is desirable to ensure that the resulting postures do not pose too high an ergonomic risk. In section 5.8 a method for introducing variation based on the concept of angle changes while ensuring low risk postures is proposed.

5.8 The Variation Method

Since variation in posture is positive and desirable, but the existing ergonomic assessment methods do not provide measures for repetitiveness or, more specifically, variation that are directly useable for ensuring variation in HRC, a new method based on angle changes is created. This section presents the method proposed for introducing variation in HRC.

Whenever variation is introduced, it will inevitably cause a deviation from the optimal. In this way, one can notice that there is a trade-off relationship between wanting the optimal posture and introducing variation. The important task when introducing variation is to find variate poses that do not deviate too much from the optimum. The limit for what is too much deviation from the optimal posture is difficult to define and probably should be specified by health care researchers or other experts. However, tools like the REBA do provide an indication of what postures are acceptable, and the postures generated with variation can thus be evaluated afterwards with REBA in order to determine whether it is an acceptable posture. Regardless of what the limit of deviation from the optimum is, the immediate challenge is how to introduce variation while minimising deviation from the optimum.

The proposed method is based on generating a sampling set around the minimum found in the optimisation in section 5.5. Since this function is strictly convex and in each dimension it is symmetric around the optimal value found in that dimension, a sampling set can be defined by

$$S = \{q \in \mathbb{R}^n | q_i^* - m_i \leq q_i \leq q_i^* + m_i\} \quad \text{for } i = 1, \ldots, n \tag{5.11}$$

Where q is the posture consisting of joint angles, q_i is the i^{th} entry in q and is the angle for the i^{th} joint, q_i^* is the i^{th} entry in q^* which is the minimum of the optimisation of the differentiable REBA in section 5.5 and is the optimal angle for the i^{th} joint, m_i is the variation margin for the i^{th} joint, and n is the amount of joints.

If S is used for sampling postures it will be known that each joint angle $\hat{q}_i$ in the sample $(\hat{q})$ will have a maximum distance to the optimal angle q_i^* of m_i. The sample set for each dimension is created by generating s_i evenly spread samples in the interval $[q_i^* - m_i, q_i^* + m_i]$, and s_i is an uneven number in order to make sure that q_i^* is one of the samples. However, the sample set for the knee angle is an exception. For this joint the sampling interval is not $[q_i^* - m_i, q_i^* + m_i]$, but rather $[q_i^*, q_i^* + m_i]$. This is due to how the knee angle is defined (see Figure 5.10) and the fact that it is physically impossible to place the knee in a negative angle.

The sampling from the created sample set happens by generating a set of n random numbers, one in each of the n intervals $[1, s_i]$. The random numbers are sampled from a uniform distribution. The set of random numbers are then used for extracting the corresponding sample value for each joint and obtain a sample $\hat{q}$ in S.

Subsequently, the distance from the sample to the k last found postures q_j for $j = 1, \ldots, k$ is examined. If the distance to all the k last postures is larger than a specified distance d, $\|\hat{q} - q_j\| \leq d$ for $j = 1, \ldots, k$, the sample is accepted as a new posture. If not all k last postures are at an acceptable distance of the sample, the sample is discarded, and the process is rerun starting with the generation of a new set of random numbers. When a new posture has been accepted, the new contact point needed for the positioning of the meat can be calculated based on the found posture using the forward kinematics of the human body presented in section 5.6. The method is summarised in the simplified flowchart in Figure 5.12, and the full python code for the method can be found in Appendix F.

The presented method has several parameters that can be adjusted according to wanted properties or ergonomic knowledge. The parameters are the tolerated margins for the deviation from the optimal posture (m_i), the number of samples in the margin interval for each joint (s_i), the required distance to the considered previous samples (d) and the amount of previous samples to evaluate distance from (k). m_i controls how far from the optimal posture samples can be generated while s_i specifies the amount of different possible sample values within the accepted interval for each joint. s_i together with m_i determines how densely the individual angle intervals are sampled. d influences the amount of variation required from posture to posture, and k determines for how many steps the space in the sampling set around a certain posture should be avoided.

The proposed method is tested in a set of evaluational tests in section 5.9.

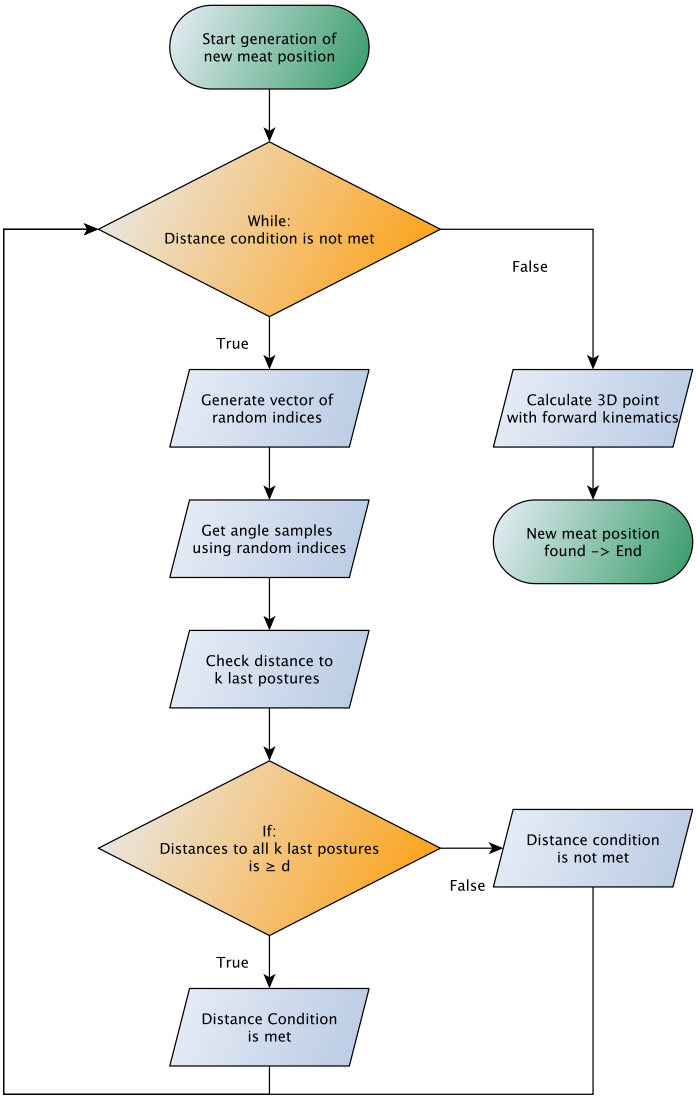


Figure 5.12: The overall structure of the variation method.

5.9 Validation of the Variation

In order to validate the performance of the proposed method and ensure that the method introduces variation in a desirable way, some test were conducted.

During the tests, the parameters mentioned in section 5.8 were set to the following values:

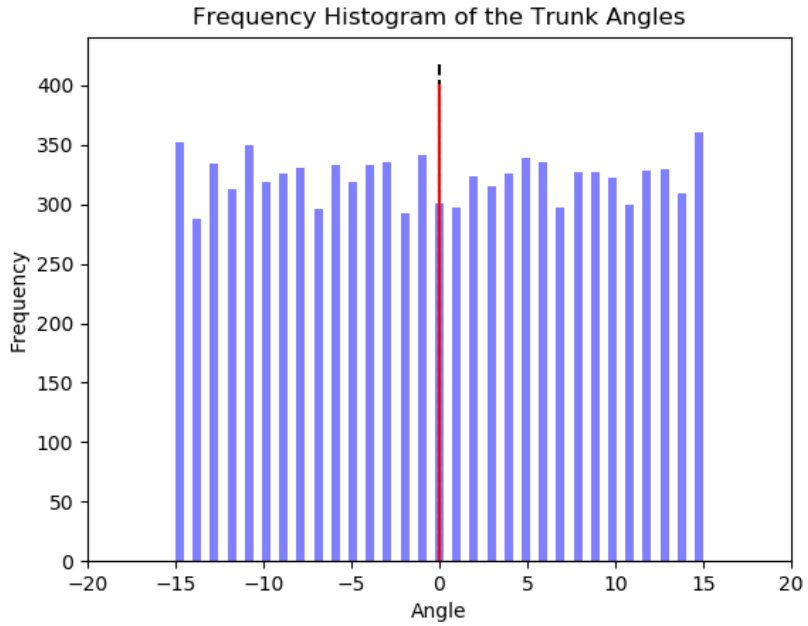
- $m_i = 15° \forall i$
- $s_i = 31 \forall i$
- $d = 25$
- $k = 5$

These values were estimated to be suitable. The value $m_i = 15°$ allows for some variation in the angles while not allowing very extreme angles. With $m_i = 15°$ and $s_i = 31$ the possible angle samples are exactly $1°$ apart which seems convenient. d is set to be 25 to not be to restrictive but still have effect during the tests. The same can be stated for the value of $k = 5$. When specifying the parameters, one should be considerate of the combined effect that they have. For

example, if d is moderate and k is very large or if k is moderate and d is very large, then the "banned area" due to too small distance to q_j for $j = 1, \ldots, k$ might cover the entire sampling set S making it impossible to find a compliant posture.

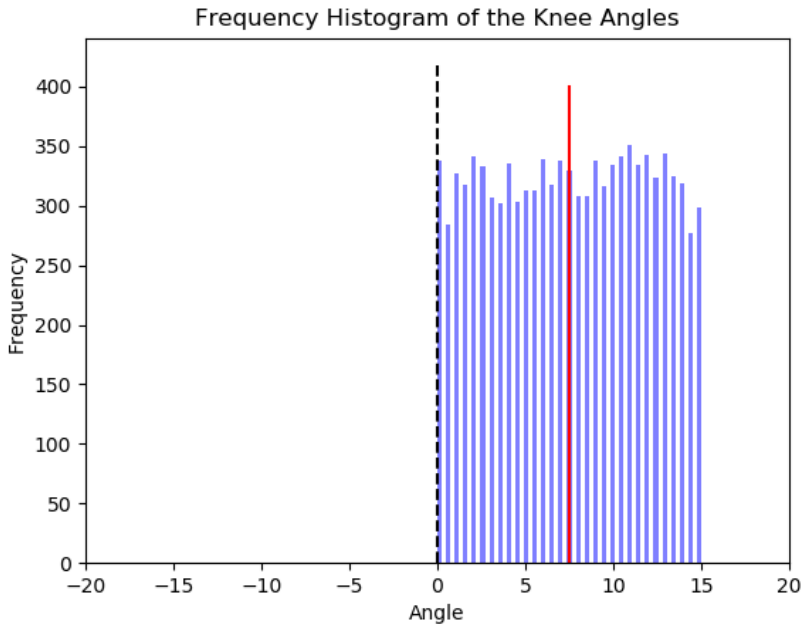
The important properties of the variation method to validate is that the generated postures in fact are distributed symmetrically around the optimal posture, and that the generated postures are ergonomically correct.

A test was run where 10.000 consecutive postures were generated in order to examine the distribution of the samples. Each sample consist of 15 angles, as there are angles for, e.g., both the right and the left arm. However, there is only one angle determined for the knees due to the assumption in the use case of standing work with both feet on the ground. This implies that the knee angle for both legs must be the same. Figure 5.13 shows the histograms of the angles for the trunk and the knees found in the 10.000 postures. Equivalent histograms for the remaining 13 joints can be found in Appendix G.



(a) Histogram of the trunk angles.

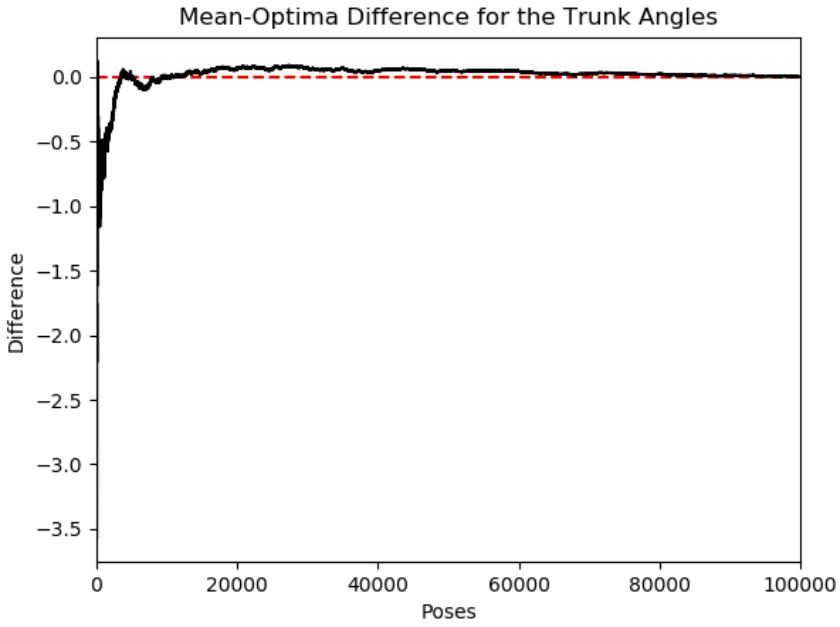
Figure 5.13: The histograms of the 10,000 angles found using the variation method presented in section 5.8. The red line indicates the mean of the sampled angles. The dotted black line indicates the optimal angle.



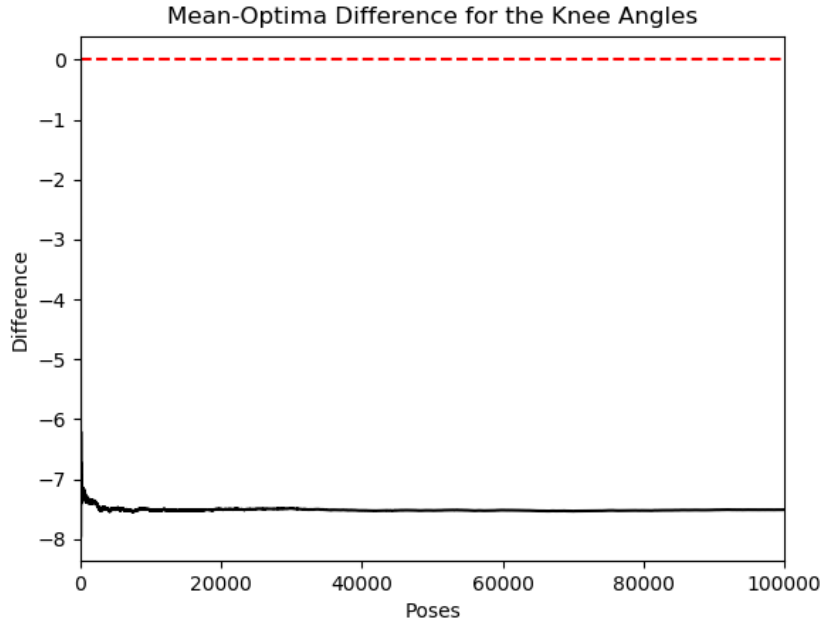
(b) Histogram of the knee angles.

Figure 5.13: The histograms of the 10,000 angles found using the variation method presented in section 5.8 (cont.). The red line indicates the mean of the sampled angles. The dotted black line indicates the optimal angle.

From the histograms in Figure 5.13 and Appendix G it can be noted how the sampled angles seem to be symmetrically distributed around the optimal angles. Furthermore, the sample mean aligns well with the optimal angle with only small deviations from the optimal angle for a few of the joints. The only clear exception is the knee angle, where there is a larger difference between the sample mean and the optimal angle due to the asymmetric sampling for this joint. Furthermore, it can be noted how the samples appear to be uniformly distributed. This is expected as this is the distribution the random numbers in the variation method are sampled from. Table 5.7 and Table 5.8 provide the mean values and the variance for the generated angles at each joint for different amounts of samples. To get a more thorough insight into how the difference between the sample means and the optimal angles develop as the number of samples increase, a test was run with 100.000 samples, where the optima-mean differences was continuously calculated. A pair of examples of the resulting plots can be seen in Figure 5.14. The plots for the remaining 13 angles can be found in Appendix H. The plots in Figure 5.14 and Appendix H indicate that as the amount of samples increase, the differences between the sample mean and the optima go towards 0, except for the mean of the knee angles, which as mentioned is a special case where the difference goes towards $\frac{m_{\text{knee}}}{2}$. In Table 5.9 the values for the optima-sample mean differences for select amounts of samples are provided. From the values in the table it can be noticed how the the optima-sample mean differences for all joints but the knees are smaller than 0.1° when considering a large amount of samples. The deviation of the sample means from the optima is thus minimal, and the sampled angles are centred around the optimal values for each joint.



(a) The optimum-sample mean difference for the trunk angles.



(b) The optimum-sample mean difference for the knee angles.

Figure 5.14: Plots of the difference between the optimal angle and the sample mean as the number of samples increases. The red line indicates the zero difference line.

Samples	$\mu_{\theta_{\mathrm{t}}}$	$\mu_{\theta_{\mathrm{ts}}}$	$\mu_{\theta_{\mathrm{n}}}$	$\mu_{\theta_{\mathrm{ns}}}$	$\mu_{\theta_{\mathrm{k}}}$	$\mu_{\theta_{\mathrm{rua}}}$	$\mu_{\theta_{\mathrm{raa}}}$	$\mu_{\theta_{\mathrm{lua}}}$	$\mu_{\theta_{\mathrm{laa}}}$	$\mu_{\theta_{\mathrm{rla}}}$	$\mu_{\theta_{\mathrm{lla}}}$	$\mu_{\theta_{\mathrm{rw}}}$	$\mu_{\theta_{\mathrm{rwd}}}$	$\mu_{\theta_{\mathrm{lw}}}$	$\mu_{\theta_{\mathrm{lwd}}}$
100	0.150	-0.1000	9.360	0.340	7.060	-0.580	-0.380	2.000	-0.910	80.280	80.670	-0.550	-0.430	0.050	0.020
1,000	0.737	-0.151	9.651	0.015	7.336	-0.106	0.150	-0.284	0.139	80.164	79.741	0.118	0.265	0.205	-0.263
10,000	0.004	0.077	9.865	-0.002	7.510	0.050	0.084	-0.119	0.021	80.048	80.073	-0.018	-0.019	-0.053	-0.057
100,000	-0.003	-0.0041	9.960	0.035	7.508	-0.017	0.032	-0.027	-0.028	80.013	79.986	0.016	-0.041	0.032	0.009

Table 5.7: The mean angles for the different joints based on different amounts of samples. The angles are stated in degrees.

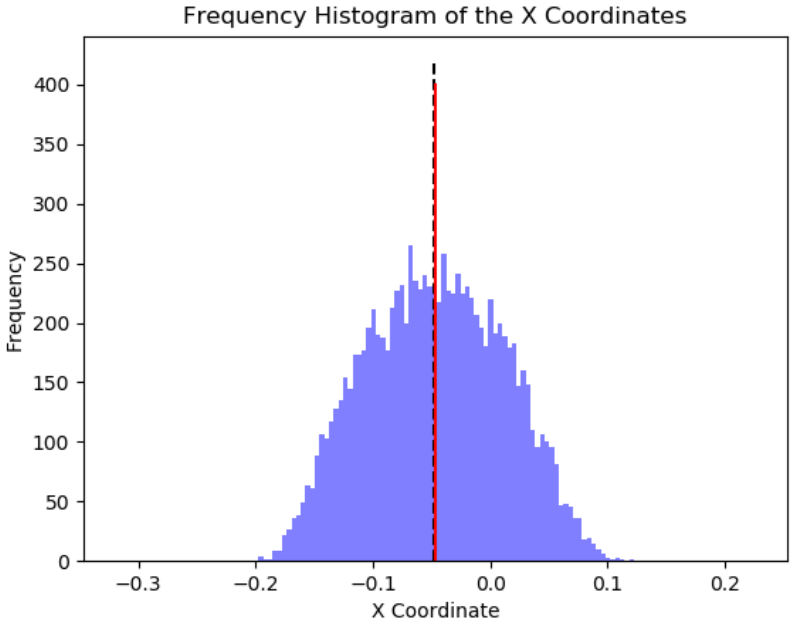
Samples	θ_t	θ_{ts}	θ_n	θ_{ns}	θ_k	θ_{rua}	θ_{raa}	θ_{lua}	θ_{laa}	θ_{rla}	θ_{lla}	θ_{rw}	θ_{rwd}	θ_{lw}	θ_{lwd}
100	84.587	65.330	70.090	73.104	20.561	88.884	71.056	71.340	84.142	82.002	78.201	82.647	80.825	81.728	75.980
1,000	80.534	75.550	80.929	77.851	19.142	82.205	74.214	81.391	78.522	75.901	78.916	81.818	77.363	77.251	76.798
10,000	80.841	78935	80.704	79.187	19.733	82.061	80.465	79.046	80.927	80.051	80.457	81.087	79.307	80.000	79.756
100,000	80.243	79.951	79.942	80.083	20.021	80.182	80.118	79.775	80.332	80.155	80.302	80.211	80.151	79.664	79.791

Table 5.8: The variance of the angles for the different joints based on different amounts of samples.

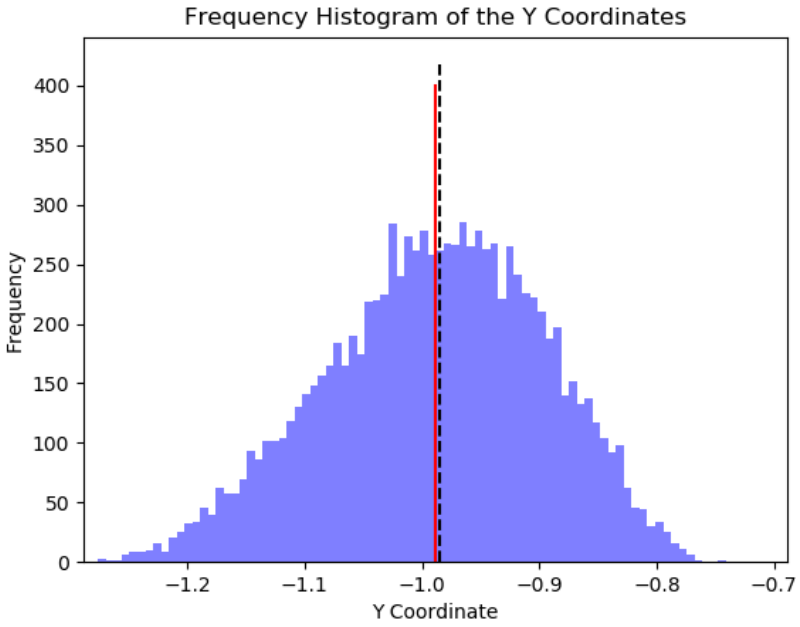
Samples	θ_t	θ_{ts}	θ_n	θ_{ns}	θ_k	θ_{rua}	θ_{raa}	θ_{lua}	θ_{laa}	θ_{rla}	$\mu_{\theta_{\mathrm{lla}}}$	θ_{rw}	θ_{rwd}	θ_{lw}	θ_{lwd}
100	-0.15	0.10	0.64	-0.34	-7.06	0.58	0.38	-2.00	0.91	-0.28	-0.67	0.55	-0.43	-0.05	-0.02
1,000	-0.74	0.15	0.35	-0.02	-7.34	0.11	-0.15	0.28	-0.14	-0.16	0.26	-0.12	-0.27	-0.21	0.26
10,000	-0.004	-0.077	0.13	0.002	-7.510	-0.050	-0.084	0.119	-0.021	-0.048	-0.073	0.018	0.019	0.053	0.057
100,000	0.003	0.041	0.040	-0035	-7.509	0.017	-0.032	0.027	0.028	-0.012	0.014	-0.016	0.041	-0.031	-0.009

Table 5.9: The optima-mean angle differences for the different joints based on different amounts of samples. The angle differences are stated in degrees.

Having examined the distribution of the sampled angles, the next step is to investigate the distribution of the resulting contact points in 3D. In Figure 5.15 the histogram plots of the x, y and z-coordinates of the found contact points are presented.



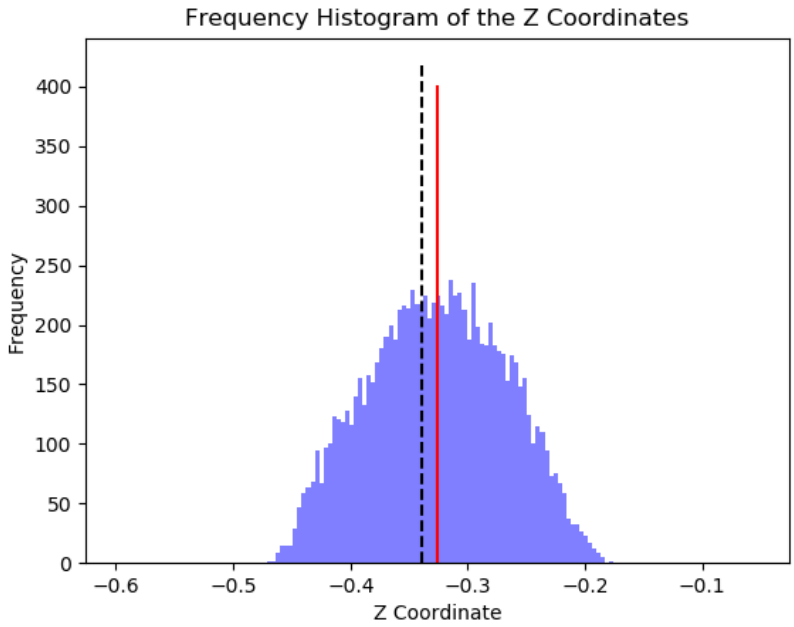
(a) The histograms of the x-coordinates of the 10,000 points.



(b) The histograms of the y-coordinates of the 10,000 points.

Figure 5.15: The histograms of the 10.000 point coordinates found using the variation method presented in section 5.8. The red line indicates the mean of the sampled point coordinates. The dotted black line indicates the given coordinate of the point found with the optimal posture.

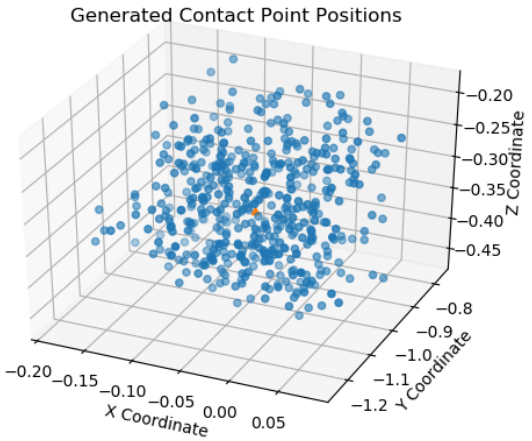
The histograms in Figure 5.15 indicate that the points obtained from the sampled postures are distributed approximately symmetrically around the coordinates for the optimal point. Further-



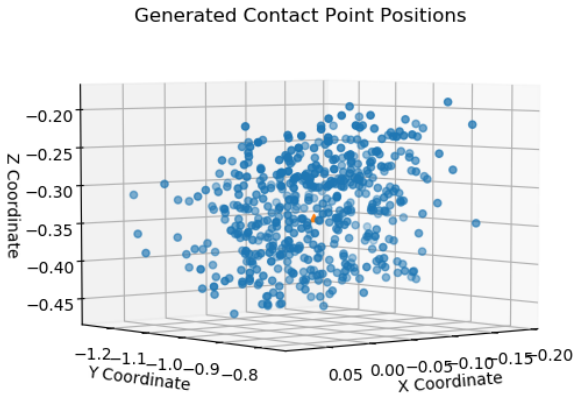
(c) The histograms of the z-coordinates of the 10,000 points.

Figure 5.15: The histograms of the 10.000 point coordinates found using the variation method presented in section 5.8 (cont.). The red line indicates the mean of the sampled point coordinates. The dotted black line indicates the given coordinate of the contact point derived from the optimal posture.

more, the coordinates of the points resemble a normal distribution. This means that a relative large portion of the uniformly sampled postures result in a point close to the optimal contact point. However, the distribution of the x-coordinates seem to be somewhat skewed and the mean of the z-coordinates does not seem to fully align with the z-coordinate of the optimal point. To get a better picture of the distribution of the derived sample points, Figure 5.16 provides a pair of 3D plots of 500 of the points as well as the optimal point.



(a) A 3D plot of 500 contact point (view a).



(b) A 3D plot of 500 contact point (view b).

Figure 5.16: 3D plots of 500 contact point calculated from the generated postures. The orange triangle marks the contact point found from the optimal posture.

The mean values and the variance of the coordinates are provided in Table 5.10 and Table 5.11, respectively. The variances recorded are all between 0.0031 and 0.0086 this means that the standard deviations are between 0.056 and 0.093 metres. If the distribution of the coordinates is assumed normal, this means that 99.7% of the coordinates will be within $\mu \pm 3 \cdot 0.056 = \mu \pm 0.168$ or $\mu \pm 3 \cdot 0.093 = \mu \pm 0.279$. However, with a normal distribution the majority of the coordinates will be close to the mean. The variances obtained based on the largest amount of samples indicate that the variance is larger for the y-coordinates compared to the x and the z-coordinates. This might be a result of the fact the majority of the joints for which angles are samples are oriented in such a way that they mainly affect the y-coordinate and to a smaller degree the other coordinates.

Samples	$\boldsymbol{\mu_x}$	$\boldsymbol{\mu_y}$	$\boldsymbol{\mu_z}$
100	-0.044	-0.984	-0.322
1,000	-0.047	-0.985	-0.327
10,000	-0.047	-0.989	-0.326
100,000	-0.046	-0.988	-0.325

Table 5.10: The mean coordinates of the found contact points based on different amounts of samples. The coordinates are stated in metres.

Samples	$\boldsymbol{x}$	$\boldsymbol{y}$	$\boldsymbol{z}$
100	0.00364	0.0086	0.0040
1,000	0.0031	0.0082	0.0036
10,000	0.0034	0.0086	0.0034
100,000	0.0034	0.0086	0.0033

Table 5.11: The variance of the coordinates of the found contact points based on different amounts of samples.

The optima-sample mean differences for the point coordinates can be found in Table 5.12. The values indicate that the difference for the x and y-coordinates is on the millimetre scale and is less than 0.5 centimetres while the difference for the z-coordinates consistently seem to be 1 and 2 centimetres. It is possible that the optimal z-coordinate simply does not lie perfectly central in the possible z-coordinates based on the sampling set.

Samples	$\boldsymbol{x}$	$\boldsymbol{y}$	$\boldsymbol{z}$
100	-0.0036	-0.0011	-0.0175
1,000	-0.0008	-0.0003	-0.0126
10,000	-0.0013	-0.0034	-0.0138
100,000	-0.0018	-0.0.0032	-0.0144

Table 5.12: The optima-mean point differences for the different coordinates based on different amounts of samples. The coordinate differences are stated in metres.

The ergonomic correctness of the generated postures has also been examined in order to validate that the postures indeed do not pose too high an ergonomic risk. The ergonomic risk was evaluated using the discrete REBA, while still disregarding the factors mentioned in section 5.4. The histogram of the resulting REBA score is shown in Figure 5.17.

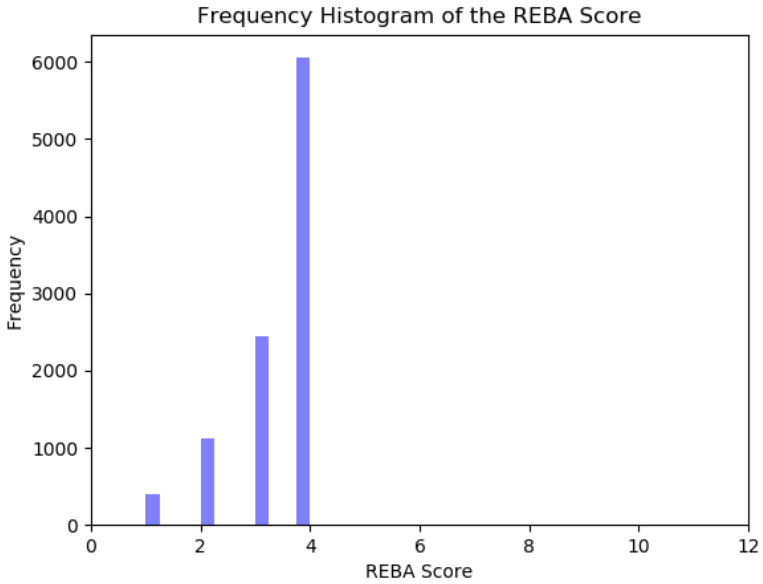


Figure 5.17: The distribution of the REBA scores of the 10,000 postures.

From Figure 5.17 it can be concluded that there is a higher frequency of the larger REBA scores. When reflecting on this, it appears somewhat logical. This might simply be because there are more possible combinations of angles that can cause the middle REBA scores in the same way that there are more ways to get 7 eyes than to get 3 eyes when rolling two die. The average REBA score for the postures found using the variation method is 3.4. The REBA method also provides guidelines for how to interpret the score levels. A score of 1 should be interpreted as "Negligible risk", while a score of 2-3 is described as "Low risk - change may be needed". A score of 4-7 is considered "medium risk - further investigation - change soon". Taking into account the fact that the proposed method indeed introduces variation or change, and the REBA scores are all in the lower end of the scale, one could argue that the levels of ergonomic risk of the generated postures are acceptable.

Besides the tests conducted in relation to validating the proposed method, a small performance tests was carried out. In respect to using the method real-time for contact point generation for the positioning of the meat pieces by a robot, it is relevant to know how fast the method can generate new points. Therefore, a test was run 10 times where 10.000 points were generated. The program finished within one minute in all runs, and the average execution time for finding 10.000 points was 56.7512 seconds. Though, the time will wary depending on the parameters chosen for the method and the computing power available, the method seems to be fast enough to use for the stepwise adaptation between the processing tasks carried out by the butcher. The test was run with the parameters specified in the beginning of this section. During the generation of the 10.000 points with these parameters, 82 samples were discarded due to insufficient distance to the previous found postures.

In section 5.10 the system advancements made in this chapter are evaluated and compared to the specified requirements revised in section 4.8.

5.10 System Evaluation

This section evaluates the advancements made in this chapter in respect to the relevant requirements.

In this cycle of the research work a new camera and a new ROS-OpenPose wrapper was introduced. The purpose of this was to gain the ability to detect the human pose in 3D. The ZED2 camera with its associated wrapper successfully provides 3D data that can be used by the ros_openpose wrapper.

The ros_openpose wrapper is able to successfully detect the human pose and provide the 3D key points. However, it has its limitations in respect to precision. Because the 3D points are found in the way they are where the 2D key points are projected onto the depth map from the ZED2 camera, the detected point will always be located on the surface of the detected human even though it might be more anatomically correct and a better representation of the humans true pose to have the points placed central in, e.g., limbs. For simple poses, this problem might only result in a small offsets while it might result in larger offsets when the posture is self occluding.

To summarise, the new setup does provide realtime non-invasive vision-based detection of the key points, and it does provide 3D capabilities in respect to pose detection. However, the precision can not be said to be within a ±2.5 margin due to the issues described above. The advanced system currently does not provide 3D adaptation capabilities as it is not yet combined with the robot. However, the 3D pose data provides the necessary information such that it can be achieved.

The work carried out with creating a differentiable REBA and the variation method ensures that a more complex and holistic ergonomic assessment is in place. Furthermore, the variation method ensures that the system complies with the requirement for inclusion of a measure for repetition.

Some of the requirements are closely related to the interplay of the rest of the system with the robot. The robot should adapt from task to task. This means that the command rate requirement and the delay requirement both change. The command rate is no longer required to be high enough for a new command always to be available. With the task to task adaptation, the important point is that a new positioning point is available when a new meat piece has to be positioned in front of the butcher. This means that the required command rate depends on the rate of the processing of meat pieces. Likewise, the requirement for the delay no longer relates to the direct realtime response to input, but rather it relates to the delay in execution of a given command or the time it takes to generate a new contact/placement point and move to that point. As seen in section 5.9, the time needed for generating 10.000 points was less than 1 minute, so it is likely that the generation time for a single point will not exceed 1 second.

The framework for the task to task adaptation is not yet implemented; however, the ergonomic optimisation method now supports that this kind of adaptation approach can be implemented.

The evaluation of the requirements is summarised in Table 5.13. The new modules developed in this chapter work together in a different way than the system overviews in Figure 5.1 and Figure 5.2 portray. Though the ergonomics of the human might be monitored, this is not used as a continuous input for the modules responsible for the adaptation. Likewise, there is no longer an isolated decision making module. The ergonomic "decisions" are contained in the proposed method for generating positioning points. The resulting new flow of the system is that when a

1	Non-invasive vision system	✓
2	Ergonomic assessment	✓
3	Repetition measure	✓
4	Adaptive behaviour from task to task	✗
5	3D capabilities	✗
6	Precision of ±2.5 cm	✗
7	Realtime monitoring	✓
8	Command Rate ≥ 3	✓
9	Delay ≤ 1 s	✗
10	Collaborative Manipulator	✗

Table 5.13: A checklist of the requirements. Note that some requirements are marked unfulfilled because the methods developed are not yet combined with the robot

meat piece should be positioned, the position and orientation of the human is obtained using the 3D pose estimation. Then a semi-optimal posture is generated using the variation method; and, subsequently, the positioning point is calculated using the information about the, position, orientation and forward kinematics of the human as well as transformation from the camera frame to the robot frame. Finally, a robot control module executes the positioning of the meat piece. The new structure of the control system is shown in Figure 5.18.

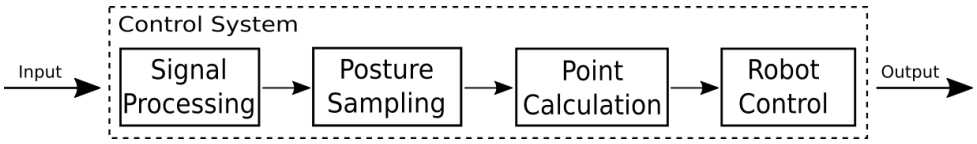


Figure 5.18: A view of the structure of the new control system.

In in the following chapter, chapter 6, aspects of the research and the proposed methods will be discussed.

Chapter 6

Discussion

In this chapter different aspects of the research work and the developed methods are discussed and related considerations are shared.

6.1 Variation Method Parameters

The proposed variation method is specified by a set of parameters: the margin sizes m_i, which specify the maximal allowed deviations from the optima; the sampling amount s_i, which specifies the amounts of different values of the joint angle q_i in the $[q_i^* - m_i, q_i^* + m_i]$ interval; the distance d, which specifies the required minimal angle distance to the k last postures; and k, which specifies the amount of previous postures evaluated with respect to distance.

During the tests in section 5.9, the parameters were specified as follows based on an attempt to estimate somewhat realistic and suitable values that did not result in too unergonomic postures.

- $m_i = 15° \forall i$
- $s_i = 31 \forall i$
- $d = 25$
- $k = 5$

However, the basis for assessing ergonomically valid values for these parameters is limited. Ideally, some investigational work should be conducted by experts within the field of ergonomics in order to determine the ergonomically correct values for the different parameters. During the tests all m_i and s_i were set as the same, however, from an ergonomic point of view they might have to be different for each joint type. Some joints might be more sensitive to angle changes than other.

The value d is currently used to find the distance from one set of angles to another. It is possible that it would be more correct to have $d = \{d_1, \ldots, d_i\}$, and examine the distances between the angles for each joint individually. Some joints might be less sensitive to repetition or longer durations in the same configuration. For those joints, maybe the required distance to the previous angle state should be smaller than for other joints, or the amount of considered previous angles for that joint k_i should be smaller.

Nonetheless, a method has been proposed that, thanks to the parameters, is flexible and can be adjusted to comply with the relevant ergonomic knowledge.

6.2 Discussion of Variation Method

The proposed method successfully generates variate postures and, thus, variate contact points. However, it samples uniformly from the sample set and does not actively seek the acceptable

sample with the minimal ergonomic score. A different approach for finding the next position could be a method that actually finds this sample.

Taking a starting point in the optimisation carried out in section 5.5, an approach could be constraint optimisation. A constraint optimisation that would introduce variation could be optimising the differentiable REBA with the constraint that the solution should have a minimum distance d to the k last points (see Equation 6.1).

$$\begin{aligned} \min_{q} \quad & \sum_{i=1}^{n} w_i Q_i(q_i) \\ \text{subject to} \quad & \|q - q_j\| \geq d \quad \text{for } j = 1, \ldots, k \end{aligned} \tag{6.1}$$

Where n is the number of joints considered in REBA, $Q_i(q)$ is a second degree polynomial of the angle q_i at joint i and w_i is the weight of $Q_i(q)$ and is a scalar. q_j is the j^{th} last found solution to the optimisation given it exists.

However, this is an optimisation of a convex function over a non-convex set and, thus, not a convex optimisation problem. This means that though optimisation of the problem might lead to some local minimum, there is no guarantee that the global minimum will be found. Essentially, the optimisation would continuously work towards minimising the variation while the constrains would be set in place to ensure variation. Also, the amount of variation would be solely dependent on the parameters k and d.

It is difficult to determine whether this optimisation approach would be preferable to the proposed method, and it is likely that the effects of the two methods could be similar dependent on the set parameters. Then, for either approach it would be a question of tweaking the parameters. Nonetheless. it is likely that the variation method proposed would requite less computations and thus also provide new positioning points faster than a constraint optimisation would.

6.3 Effect of the Adaptive and Variation Introducing Methods

Methods for finding an ergonomically optimal placement of an object by a robot that a human should interact with have been presented in chapter 4 and chapter 5.

The adaptive method applied in the IS indeed continuously adapt the position of the object with the purpose of guiding the human to an optimal posture, but the success of the method rests on the assumption that the adaptive moves will introduce a certain change in the posture of the human. If this is not true, and the human adjusts to the change in a different way than expected, the method might never reach the truly optimal position, or might even "wander off" to a very inappropriate position.

For the optimisation method based on the REBA, it is certain that the optimal position given the ergonomic relations reflected in REBA can be found. But even then, the human body has many mechanical Degrees of Freedom (DOF) and there is no guarantee that the human interacting with the object chooses exactly the found optimal posture from all the poses that result in the hand being positioned at the object.

This challenge of the many DOF of the human body, and the humans freedom to chose between the postures that ensure contact is a central and unavoidable issue when attempting to affect the posture of a human through the positioning of a contact point or an interaction object. This concern also affected the choices made when constructing the variation method. For example, a key decision to make was whether to evaluate the distance to the k last samples in the joint angle space or the cartesian space of the resulting contact points. The choice was made to evaluate in joint angle space because it is the joint angles that essentially should be varied. However, it can easily be argued that it is unknown whether the human choses the

sampled posture to reach the resulting contact point. Conversely, if the Euclidean distance to the k last contact points is evaluated, it can be argued that the change in position only has minimal effect on the posture.

Conclusively, it can be stated that in order to validate the effect of the proposed method and get a better understanding of the effect of the position of a contact point on the posture, user testing should be conducted. During the tests, the contact point should be varied and the pose of the test subjects should be recorded in order to examine the body angles and the variance introduced in these angles. Only then will it be known if the method is truly effective and to which extent repetitiveness has been reduced. It would, also, be interesting to compare the comfort perceived by test subjects repeating a certain HRC task for an extended period of time with and without the variation method.

6.4 Possible Advancements

This section describes a selection of possible advancements or new approaches that could be the next step in improving the implemented system at its current state or the next step in the exploration of adapting robotic movements based on ergonomics.

6.4.1 A Combined System

The most obvious advancement to make of the system in its current state would be to combine the advancements described in chapter 5 with the unchanged parts of the system from the IS. This would enable the robot to actually place a simulated meat piece at the generated cartesian point in respect to a human.

In order to combine the parts of the system and make it function in the desired way, a few things are needed. Firstly, the camera should be calibrated in respect to the robot, such that the points detected by the camera can be obtained in the robots 3D space. Secondly, calculations in order to obtain the orientation of the human in the 3D space of the camera is needed. A framework for handling the task by task adjustment also needs to be set in place. And lastly, for each subject the robot should adjust according to, the length of the body segments are needed. These should either be determined beforehand and provided to the adaptive system or a module for detecting them should be added to the system.

This combination of the system elements would be a valuable next step and this is also needed if the user testing described in section 6.3 is to be carried out.

6.4.2 Cumulative Measure for Repetitiveness

A possible improvement of the system could be to consider the accumulated effect of postures caused by a certain placement of the meat piece rather than simply choosing a new position for each processing operation.

This could for example be done by using the same position until an accumulated, .e.g. REBA, score for either the generated or the detected postures reaches a certain limit. Once the limit is met then a new positioning of the meat could be chosen. Naturally, postures resulting in higher scores will contribute more to the accumulation and will cause the accumulated score to reach the limit within fewer postures. This means that positions of the meat that cause higher risk postures only would be reused relatively few times while positions that cause low risk postures can be reused more times.

A method like the described could help prevent unnecessary variation; however, by definition such a procedure would decrease the variation otherwise introduced by the variation method proposed in section 5.8. Essentially, the addition of this procedure would introduce yet another

parameter for experts to specify in order to make the behaviour of the method comply with knowledge about ergonomic relationships, namely, the accumulation limit.

6.4.3 Optimisation Across Cutting Line

In the proposed method the optimisation happens only for one contact point. However, in relation to the cutting operation carried out by the butcher in the use case of this project, it might be more correct to optimise in respect to the cutting motion or cutting line. Such an optimisation might consider ergonomic score of only the butcher's posture at the end points of the cutting line, or it might include the ergonomic state at several via-points along the cutting line. Such an optimisation might also be useful for finding the optimal orientation of the meat piece such that the ergonomics of performing the cutting motion are optimised. However, creating this optimisation is complex and requires knowledge of tool handling and inverse kinematics. This will also be a non-convex problem, which again makes it impossible to guarantee that the global optimum is found.

Chapter 7

Conclusion

The research work presented has been focussed on exploring the possibility of adapting robotic movements in Human-Robot Collaboration (HRC) ergonomically according to vision-based input. The research was conducted in the context of a use case defined within the meat packing industry.

During the work an Initial System (IS) has been implemented that adapts according to posture and serves as a concept demonstrator. The IS has subsequently been subject to further advancements in relation to the central focus of the research work specified by the hypothesis;

Through the use of ergonomic measures and constrains it is possible to optimise the ergonomic state of a human collaborating with a robot as well as minimise exposure to repetitiveness.

A method has been proposed for generating points for positioning of an objet by a robot that is to be processed by a human collaborator. The method takes both posture and receptiveness into account and ensures successful generation of semi-optimal variated placement points for the robot. While a maximum tolerated ergonomic score for the posture can be specified, the method does not find the global solution with the lowest known score for any given set of constraints.

In order to ensure and validate that the proposed method truly fulfils its purpose, user testing should be conducted. Only through testing can the true effect of the generated contact/placement points on the ergonomic state of the human collaborator be revealed, and, thus, the hypothesis be confirmed.

On a final note, it can be concluded that ergonomic adaptation of robotic movements is a field with many unknowns and unsolved challenges that needs further exploration and research.

List of Figures

List of Tables

Bibliography

Abobakr, A., Nahavandi, D., Iskander, J., Hossny, M., Nahavandi, S., and Smets, M. A kinect-based workplace postural analysis system using deep residual networks. In *2017 IEEE International Systems Engineering Symposium (ISSE)*, pp. 1–6, 10 2017. doi: 10.1109/SysEng.2017.8088272.

Arbejdstilsynet, 2018. *ARBEJDSTILSYNET ÅRSOPGØRELSE 2017 - ANMELDTE ERHVERVSSYGDOMME 2012-2017 - Bilag B: Supplerende tabeller og figurer.*

Bhattacharya, A., Greathouse, L., Warren, J., Li, Y., Dimov, M., Applegate, H., Stinson, R., and Lemasters, G., 1997. An ergonomic walkthrough observation of carpentry tasks: a pilot study. *Applied occupational and environmental hygiene*, 12(4), pp. 278–287.

Bjarne Thomsen (Chef consultant), Tommy Jensen (Federal consultant), Per Hansen (Federal secretary). 2019.

Bonfiglioli, R., Mattioli, S., Fiorentini, C., Graziosi, F., Curti, S., and Violante, F. S., 2007. Relationship between repetitive work and the prevalence of carpal tunnel syndrome in part-time and full-time female supermarket cashiers: a quasi-experimental study. *International archives of occupational and environmental health*, 80(3), pp. 248–253.

Bonnet, V., Mazzà, C., Fraisse, P., and Cappozzo, A. An optimization algorithm for joint mechanics estimate using inertial measurement unit data during a squat task. In *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 3488–3491. IEEE, 2011.

Books, H., 1998. Manual Handling Operations Regulations. 1992 Guidance on Regulations L23.

Books, H., 2002. Upper limb disorders in the workplace.

Bortot, D., Born, M., and Bengler, K., 2013. Directly or on detours? How should industrial robots approximate humans? *2013 8th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, 3. doi: 10.1109/HRI.2013.6483515.

Brown, T., James, S. J., and Purnell, G. L., 2005. Cutting forces in foods: experimental measurements. *Journal of food engineering*, 70(2), pp. 165–170.

Browne, D. *Introduction to Anatomy.* OpenStax CNX, January 2015. Available at: <http://cnx.org/contents/ca77df63-d40f-4b9d-948d-fa3f87694b31@1.1.>.

Busch, B., Maeda, G., Mollard, Y., Demangeat, M., and Lopes, M. Postural optimization for an ergonomic human-robot interaction. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 2778–2785, 2017. ISBN 1-5386-2683-7. doi: 10.1109/IROS.2017.8206107.

Busch, B., Toussaint, M., and Lopes, M. Planning Ergonomic Sequences of Actions in Human-Robot Interaction. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1916–1923, 2018. ISBN 1-5386-3082-6. doi: 10.1109/ICRA.2018.8462927.

Cao, Z., Simon, T., Wei, S.-E., and Sheikh, Y. Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields. In *CVPR*, 2017.

Chander, D. S. and Cavatorta, M. P., 2017. An observational method for postural ergonomic risk assessment (PERA). *International Journal of Industrial Ergonomics*, 57, pp. 32–41.

CMU Perceptual Computing Lab, 6 2019. *OpenPose: Real-time multi-person keypoint detection library for body, face, hands, and foot estimation.* Available at: <https://github.com/CMU-Perceptual-Computing-Lab/openpose>.

Council, E., 2004. Regulation (EC) No 853/2004 of the European parliament and of the council of 29 April 2004 laying down specific hygiene rules for food of animal origin. *J. Eur. Union*, 139, p. 32.

Courreges, F., Edkie, A., Poisson, G., and Vieyres, P. Ergonomic mouse based interface for 3D orientation control of a tele-sonography robot. In *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 61–66, 2009. ISBN 1-4244-3803-9. doi: 10.1109/IROS.2009.5354366.

Crainic, M. F. and Preitl, S. Ergonomic operating mode for a robot arm using a gamepad with two joysticks. In *2015 IEEE 10th Jubilee International Symposium on Applied Computational Intelligence and Informatics*, pp. 167–170, 2015. ISBN 1-4799-9912-1. doi: 10.1109/SACI.2015.7208192.

David, G., 2005. Ergonomic methods for assessing exposure to risk factors for work-related musculoskeletal disorders. *Occupational medicine*, 55(3), pp. 190–199.

Descatha, A., Roquelaure, Y., Chastang, J.-F., Evanoff, B., Cyr, D., and Leclerc, A., 2009. Description of outcomes of upper-extremity musculoskeletal disorders in workers highly exposed to repetitive work. *The Journal of Hand Surgery*, 34(5), pp. 890–895.

Dugan, S. A. and Bhat, K. P., 2005. Biomechanics and analysis of running gait. *Physical Medicine and Rehabilitation Clinics*, 16(3), pp. 603–621.

Eliasson, K., Palm, P., Nyman, T., and Forsman, M., 2017. Inter-and intra-observer reliability of risk assessment of repetitive work without an explicit method. *Applied ergonomics*, 62, pp. 1–8.

Farrell, T. R. and Weir, R. F., 2007. The optimal controller delay for myoelectric prostheses. *IEEE Transactions on neural systems and rehabilitation engineering*, 15(1), pp. 111–118.

Filippeschi, A., Schmitz, N., Miezal, M., Bleser, G., Ruffaldi, E., and Stricker, D., 2017. Survey of motion tracking methods based on inertial sensors: A focus on upper limb human motion. *Sensors*, 17(6), p. 1257.

Fødevareforbundet NNF and DI, 2017. *Overenskomst og aftaler - mellem Fødevareforbundet NNF og DI gældende for slagteområdet (2017-2020).*

Gorsche, R. G., Wiley, J. P., Renger, R. F., Brant, R. F., Gemer, T. Y., and Sasyniuk, T. M., 1999. Prevalence and incidence of carpal tunnel syndrome in a meat packing plant. *Occupational and Environmental Medicine*, 56(6), pp. 417–422.

Grandjean, E. and Hünting, W., 1977. Ergonomics of posture—review of various problems of standing and sitting posture. *Applied ergonomics*, 8(3), pp. 135–140.

Grant, K. A. and Habes, D. J., 1997. An electromyographic study of strength and upper extremity muscle activity in simulated meat cutting tasks. *Applied Ergonomics*, 28(2), pp. 129–137. ISSN 0003-6870. doi: 10.1016/S0003-6870(96)00049-X.

Gupta, V., February 2019. *Pose Detection Comparison: wrnchAI vs OpenPose.* Available at: <https://www.learnopencv.com/pose-detection-comparison-wrnchai-vs-openpose/>.

Hägg, G. M., Vogel, K., Karltun, J., and McGorry, R. Knife force differences when cutting meat at different temperatures. In *NES2012 Ergonomics for Sustainability and Growth*. KTH Royal Institute of Technology, 2012.

Hignett, S. and McAtamney, L., 2000. Rapid entire body assessment (REBA). *Applied ergonomics*, 31(2), pp. 201–205.

Högberg, D., Brolin, E., and Hanson, L. Concept of formalized test procedure for proactive assessment of ergonomic value by digital human modelling tools in lean product development. In *International Conference on Applied Human Factors and Ergonomics*, pp. 425–436. Springer, 2017.

iPi Soft, 6 2019. Available at: <http://ipisoft.com/store/>.

Ketola, R., Toivonen IV, R., and Viikari-Juntura, E., 2001. Interobserver repeatability and validity of an observation method to assess physical loads imposed on the upper extremities. *Ergonomics*, 44(2), pp. 119–131.

Kim, W., Lorenzini, M., Balatti, P., Nguyen, P. D., Pattacini, U., Tikhanoff, V., Peternel, L., Fantacci, C., Natale, L., Metta, G., et al., 2019. Adaptable workstations for human-robot collaboration: A reconfigurable framework for improving worker ergonomics and productivity. *IEEE Robotics & Automation Magazine*, 26(3), pp. 14–26.

Kong, Y.-K., Han, J.-G., and Kim, D.-M., 2010. Development of an ergonomic checklist for the investigation of work-related lower limb disorders in farming-ALLA: agricultural lower-limb assessment. *Journal of the Ergonomics Society of Korea*, 29(6), pp. 933–942.

Kong, Y.-K., Lee, S.-y., Lee, K.-S., and Kim, D.-M., 2018. Comparisons of ergonomic evaluation tools (ALLA, RULA, REBA and OWAS) for farm work. *International journal of occupational safety and ergonomics*, 24(2), pp. 218–223.

Kroemer, K., 1989. Cumulative trauma disorders: their recognition and ergonomics measures to avoid them. *Applied ergonomics*, 20(4), pp. 274–280.

Lee, J., Chai, J., Reitsma, P. S. A., Hodgins, J. K., and Pollard, N. S. Interactive control of avatars animated with human motion data. In *Interactive control of avatars animated with human motion data*, volume 21, pp. 491–, 2002. ISBN 1-58113-521-1. doi: 10.1145/566570.566607.

Li, G. and Buckle, P. Evaluating Change in Exposure to Risk for Musculoskeletal Disorders—A Practical Tool. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, volume 44, pp. 5–407. SAGE Publications Sage CA: Los Angeles, CA, 2000.

Logitech, 2019. *Logitech Quickcam E 3500 Specs*. Available at: <https://support.logitech.com/en_us/product/quickcam-e-3500/specs>. (visited on 29-05-19).

Lorenzini, M., Kim, W., De Momi, E., and Ajoudani, A. A New Overloading Fatigue Model for Ergonomic Risk Assessment with Application to Human-Robot Collaboration. In *2019 International Conference on Robotics and Automation (ICRA)*, pp. 1962–1968. IEEE, 2019.

Lynch, K. M. and Liu, C. Designing motion guides for ergonomic collaborative manipulation. In *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065)*, volume 3, pp. 2709–2715. IEEE, 2000.

Marcon, M., Pispero, A., Pignatelli, N., Lodi, G., and Tubaro, S. Postural assessment in dentistry based on multiple markers tracking. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1408–1415, 2017.

McAtamney, L. and Corlett, E. N., 1993. RULA: a survey method for the investigation of work-related upper limb disorders. *Applied ergonomics*, 24(2), pp. 91–99. ISSN 0003-6870. doi: 10.1016/0003-6870(93)90080-S.

Monnington, S., Quarrie, C., Pinder, A., and Morris, L., 2003. Development of manual handling assessment charts (MAC) for health and safety inspectors. *Contemporary Ergonomics*, pp. 3–8.

Moore, J. S. and Garg, A., 1995. The strain index: a proposed method to analyze jobs for risk of distal upper extremity disorders. *American Industrial Hygiene Association Journal*, 56(5), pp. 443–458.

Moore, J. S. and Garg, A., 1998. The effectiveness of participatory ergonomics in the red meat packing industry Evaluation of a corporation. *International Journal of Industrial Ergonomics*, 21(1), pp. 47–58. ISSN 0169-8141. doi: 10.1016/S0169-8141(97)00024-3.

Nvidia, 2020. *Nvidia GeForce GTX 1080 Ti*. Available at: <https://www.nvidia.com/en-in/geforce/products/10series/geforce-gtx-1080-ti/>. (visited on 22-05-20).

Obdrzalek, S., Kurillo, G., Ofli, F., Bajcsy, R., Seto, E., Jimison, H., and Pavel, M. Accuracy and robustness of Kinect pose estimation in the context of coaching of elderly population. In *2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, volume 2012, pp. 1188–1193, 2012. ISBN 1-4244-4119-6. doi: 10.1109/EMBC.2012.6346149.

Occhipinti, E., 1998. OCRA: a concise index for the assessment of exposure to repetitive movements of the upper limbs. *Ergonomics*, 41(9), pp. 1290–1311.

Papandreou, G., Zhu, T., Kanazawa, N., Toshev, A., Tompson, J., Bregler, C., and Murphy, K. P., 2017. Towards Accurate Multi-person Pose Estimation in the Wild. *CoRR*, abs/1701.01779. Available at: <http://arxiv.org/abs/1701.01779>.

Park, T. J., Tong-Jin, P., Chang-Soo, H., Jung-Soo, H., and Ho-Gil, L. Development of the gait assistant mobile robot using ergonomic design. In *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA'04. 2004*, volume 2, pp. 2037–2042. IEEE, 2004. ISBN 0-7803-8232-3. doi: 10.1109/ROBOT.2004.1308123.

Pearce, M., Mutlu, B., Shah, J., and Radwin, R., 2018. Optimizing makespan and ergonomics in integrating collaborative robots into manufacturing processes. *IEEE transactions on automation science and engineering*, 15(4), pp. 1772–1784.

Peternel, L., Kim, W., Babič, J., and Ajoudani, A. Towards ergonomic control of human-robot co-manipulation and handover. In *2017 IEEE-RAS 17th International Conference on Humanoid Robotics (Humanoids)*, pp. 55–60. IEEE, 2017.

Pontonnier, C., de Zee, M., Samani, A., Dumont, G., and Madeleine, P., 2014. Strengths and limitations of a musculoskeletal model for an analysis of simulated meat cutting tasks. *Applied ergonomics*, 45(3), pp. 592–600. ISSN 0003-6870. doi: 10.1016/j.apergo.2013.08.003.

Punnett, L. and Wegman, D. H., 2004. Work-related musculoskeletal disorders: the epidemiologic evidence and the debate. *Journal of electromyography and kinesiology*, 14(1), pp. 13–23.

Rethink Robotics, 2019. *Sawyer*. Available at: <https://www.rethinkrobotics.com/sawyer>. (visited on 29-05-19).

Roman-Liu, D., 2014. Comparison of concepts in easy-to-use methods for MSD risk assessment. *Applied ergonomics*, 45(3), pp. 420–427.

Sanchez-Lite, A., Garcia, M., Domingo, R., and Angel Sebastian, M., 2013. Novel Ergonomic Postural Assessment Method (NERPA) Using Product-Process Computer Aided Engineering for Ergonomic Workplace Design. *PloS one*, 8(8), pp. e72703–. ISSN 1932-6203. doi: 10.1371/journal.pone.0072703.

Schaub, K. G., Mühlstedt, J., Illmann, B., Bauer, S., Fritzsche, L., Wagner, T., and Bullinger-Hoffmann, A. C., 2012. Ergonomic assessment of automotive assembly tasks with digital human modelling and the 'ergonomics assessment worksheet'(EAWS). *Int. J. Human Factors Modelling and Simulation*, 3(3/4).

Shafti, A., Ataka, A., Lazpita, B. U., Shiva, A., Wurdemann, H. A., and Althoefer, K. Real-time robot-assisted ergonomics. In *2019 International Conference on Robotics and Automation (ICRA)*, pp. 1975–1981. IEEE, 2019.

Snook, S. and Ciriello, V., 1991. Liberty Mutual Tables for Lifting, Carrying, Pushing and Pulling. *Ergonomics*, 34(9), pp. 1197–1213.

Stereolabs, 2020. *Meet ZED 2*. Available at: <https://www.stereolabs.com/zed-2/>. (visited on 01-06-20).

Sun, K., Xiao, B., Liu, D., and Wang, J., 2019. Deep high-resolution representation learning for human pose estimation. *arXiv preprint arXiv:1902.09212*.

T. C. Tan, J. and Arai, T. Triple Stereo Vision System for Safety Monitoring of Human-Robot Collaboration in Cellular Manufacturing. In *Assembly and Manufacturing (ISAM), 2011 IEEE International Symposium on*, pp. 1–6. IEEE, 5 2011.

Tappin, D. C., Bentley, T. A., and Vitalis, A., 2008. The role of contextual factors for musculoskeletal disorders in the New Zealand meat processing industry. *Ergonomics*, 51(10), pp. 1576–1593. ISSN 0014-0139. doi: 10.1080/00140130802238630.

Tensorflow, 5 2018. *Real-time Human Pose Estimation in the Browser with Tensorflow.js*. Available at: <https://medium.com/tensorflow/real-time-human-pose-estimation-in-the-browser-with-tensorflow-js-7dd0bc881cd5>.

Toxverd, F., 11 2018. *Hvert andet medlem af Fødevareforbundet NNF tager medicin for at klare arbejdsdagen*. Available at: <https://www.nnf.dk/nyheder/2018/november/hver-andet-medlem-af-fodevareforbundet-nnf-tager-medicin-for-at-klare-arbejdsdagen/>.

Ubisoft. *Just Dance*. Available at: <https://www.onmsft.com/news/just-dance-2019-is-now-live-on-xbox-one-360-consoles-with-kinect-functionality>.

van den Broek, M. K. and Moeslund, T. B. Ergonomic Adaptation of Robotic Movements in Human-Robot Collaboration. In *Companion of the 2020 ACM/IEEE International Conference on Human-Robot Interaction*, pp. 499–501, 2020.

Vicon, 6 2019. *Vero*. Available at: <https://www.vicon.com/products/camera-systems/vero>.

Vignais, N., Miezal, M., Bleser, G., Mura, K., Gorecky, D., and Marin, F., 2013. Innovative system for real-time ergonomic feedback in industrial manufacturing. *Applied ergonomics*, 44(4), pp. 566–574. ISSN 0003-6870. doi: 10.1016/j.apergo.2012.11.008.

Vink, P., Konijn, I., Jongejan, B., and Berger, M. Varying the office work posture between standing, half-standing and sitting results in less discomfort. In *International Conference on Ergonomics and Health Aspects of Work with Computers*, pp. 115–120. Springer, 2009.

Wang, K. J., Zheng, C. Y., and Mao, Z. H. Human-Centered, Ergonomic Wearable Device with Computer Vision Augmented Intelligence for VR Multimodal Human-Smart Home Object Interaction. In *Human-Centered, Ergonomic Wearable Device with Computer Vision Augmented Intelligence for VR Multimodal Human-Smart Home Object Interaction*, pp. 767–768. IEEE, 2019. ISBN 1-5386-8556-6. doi: 10.1109/HRI.2019.8673156.

Wang, Q., Kurillo, G., Ofli, F., and Bajcsy, R. Evaluation of Pose Tracking Accuracy in the First and Second Generations of Microsoft Kinect. In *Healthcare Informatics (ICHI), 2015 International Conference on*, pp. 380–389, 2015. ISBN 978-1-4673-9548-9. doi: 10.1109/ICHI.2015.54.

Waters, T. R., Putz-Anderson, V., Garg, A., and Fine, L. J., 1993. Revised NIOSH equation for the design and evaluation of manual lifting tasks. *Ergonomics*, 36(7), pp. 749–776.

Waters, T. R., Putz-Anderson, V., and Garg, A., 1994. Applications manual for the revised NIOSH lifting equation.

Xia, S., Gao, L., Lai, Y.-K., Yuan, M.-Z., and Chai, J., 2017. A survey on human performance capture and animation. *Journal of Computer Science and Technology*, 32(3), pp. 536–554.

Yazdanirad, S., Khoshakhlagh, A. H., Habibi, E., Zare, A., Zeinodini, M., and Dehghani, F., 2018. Comparing the effectiveness of three ergonomic risk assessment methods–RULA, LUBA, and NERPA–to predict the upper extremity musculoskeletal disorders. *Indian journal of occupational and environmental medicine*, 22(1), pp. 17–21. ISSN 0973-2284. doi: 10.4103/ijoem.IJOEM_23_18.

Yeung, K.-Y., Kwok, T.-H., and Wang, C. C., 2013. Improved skeleton tracking by duplex kinects: A practical approach for real-time applications. *Journal of Computing and Information Science in Engineering*, 13(4), p. 041007.

Zhang, Y., Narayanan, V., Chakraborti, T., and Kambhampati, S. A human factors analysis of proactive support in human-robot teaming. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 3586–3593. IEEE, 2015.

Zhang, Y., Sreedharan, S., Kulkarni, A., Chakraborti, T., Zhuo, H. H., and Kambhampati, S. Plan explicability and predictability for robot task planning. In *2017 IEEE international conference on robotics and automation (ICRA)*, pp. 1313–1320. IEEE, 2017.

Credit Notice & Disclaimer

This project has not been subject to any sponsorship or endorsement, nor does the project have any affiliation with any of the mentioned brands and companies mentioned in this report, including Logitech[ő].

Appendix A

The process of creating the Dockerfile

Installing software through Dockerfiles introduces several challenges which has to be solved using a selection of workarounds. One of such is confirmation og dialog windows. When a Docker image is build from a Dockerfile, the process has to run undisturbed, and if a dialog window is generated, the process dies. This might be overcome simply by using flags that deactivates the confirmation dialogs and simply assume accept such as the *-y* flag when installing packages using Apt. Closely related but more complex is the challenge of software with Graphical User Interface (GUI)s which might be needed for configuration. This is the case if you follow the standard installation guide for OpenPose where it is suggested that preinstallation configuration is done in the CMake GUI. Luckily, it is fairly easy to overcome in this case by using regular command line CMake with different flags; but if one is unfamiliar with this method, it takes time to figure out how to use it.

An entirely different challenge is when the installation or configuration requires editing of lines in files. Under normal conditions one would open the file in a text editor with some user interface; however, since this is not possible during the install process run by Docker an alternate route is needed. For this purpose the GNU stream editor (sed) is very useful. sed is a command-line tool for editing lines in files. In this project it was mainly used to edit commands in sh files or path specifications. The last general challenge worth mentioning is issues related to permissions. Some programmes or actions require special permissions which, on a normal system, would be executed using sudo. Nonetheless, since sudo is not available in Docker environments a workaround by using, eg., chmod is necessary. The use of chmod instead of sudo is one of the modifications which can be done using sed and that is often needed in sh files.

Due to the way Docker and the containers work, there are some considerations that might be valuable to make before implementing a system in Docker. What is written in a Dockerfile will always be present in any container that is an instance of the given image. On the contrary, all changes made to the system and data generated and saved on the system in a container are specific to that container and once the container is closed the content will be lost. Docker offers a linkage between directories in the host system and directories in the Docker container which can be used to overcome this property. Then the content one wants to preserve is saved in a directory on the host system which can be loaded in each time a container is run. With the characteristics of the different ways to access and combine the system with different programmes and file content it is worth making some design considerations in respect to functionality before choosing what is included in the Dockerfile and what is loaded in from a host directory. A git repository containing some software, for example, can be loaded both ways. If the software is to be used in a specific state each time the container is run, it is beneficial to include it in the

system setup in the Dockerfile. If, on the other hand, that same software is to undergo iterative changes continuously through the work, it might be preferable to store the software with the changes in an imported host directory. In this project ROS and OpenPose along with standard software and tools such as CUDA, python, vim etc. are included in the Dockerfile for the IS while the OpenPose-ROS wrapper and the ROS nodes implemented are stored in a host directory.

As mentioned earlier, the scripting of a Dockerfile can be very time consuming. This is due to the fact that the compilation goes through several steps each time a Dockerfile is compiled. As the steps often are installation of a software each step is costly and the compilation of a system can continue for the duration of half an hour or more. Though the compilation utilises cache memory to reuse previously compiled steps, changes to the Dockerfile often introduces recompilation of an extensive part of the file. Also, if a time consuming step fails at the end this will be a large amount of time spend compiling something that, in the end, is uncached. During the development of the IS a lot of time was spend debugging and recompiling the Dockerfile, a file which in its final state takes 15 to 20 minutes to compile. Even when a seemingly working system was achieved, issues inside the container with interfacing between OpenPose and ROS forced a new round of scripting and debugging of the Dockerfile. The issue encountered was that OpenPose-ROS Wrapper crashed each time a proper input image was received. Through extensive searches on the internet a solution for a different issue was found and tested as a last resort. The argumentation was that OpenPose had to be build with a linkage to the same OpenCV as the one used by ROS. In order to do this, the install order in the Dockerfile had to be changed, which resulted in more time consuming compilations. And even then, it was not working with the current install method used for the OpenPose which introduced a few hours of work implementing a different install method and debugging the process. Eventually, a container with the wanted setup working was achieved.

Appendix B

The process of establishing connection between a Docker container and the Sawyer robot

Because no communication could be established with the robot when running the Docker container with the `-net host` flag, the connection was examined by pinging to the robot's IP address. As this was possible the IP address was set up in the mentioned script instead of the robot host name. Subsequently, the container could seemingly partially access the network as it was able to see ROS topics published by the robot but could not communicate with the topics an a full two-way communication. In an attempt to fix the problem, the Docker functionality of exposing and mapping ports and specifying the container's IP address was tested without success. Similarly, settings in the Sawyer robot of how it would express itself on the network was changed without any effect.

Other researchers have been able to connect to different robots from a similar setup with ROS in a container or connecting to the Sawyer in a setup without Docker. Therefore, the assumption was that the problem was some where with Docker, and the next approach was to isolate the problem origin. In an attempt to examine whether the problem was the Docker setup, a similar system was implemented in Virtual Box. This was chosen as the Intera SDK requires a version of ROS which is incompatible with the host OS on the desktop. In Virtual Box the connection between the robot and the virtual environment could be established without any issues; thus, the problem was indeed related to Docker.

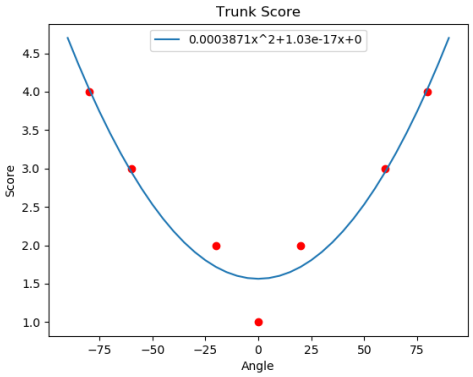
Experiments with Docker network were made. A bridge type network was created under the assumption that it would resemble the way connection was established with Virtual Box. However, this and an alternative network type, macvlan, were tested and both did not solve the issue. Throughout all of the attempts and small changes made, a variation of trials were made in the configuration of the Intera SDK to check for solutions.

Through extensive searches on the internet, a detail about Docker having its own namespace even when connected to host network was discovered. Simultaneously, a git repository with work with the Sawyer robot inside a Docker container was found. In the Docker run commands used in the git repository a `-add-host` flag was used for mapping a host name in the Docker container to an IP address. When this flag with a mapping of the Sawyer host name to its IP address was added to the run command the problem was solved and full connection between Sawyer and the container could be established.

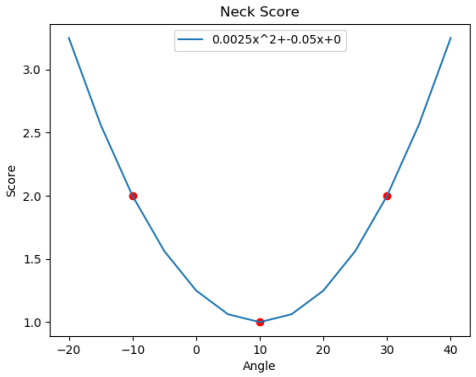
Appendix C

The second degree polynomial regressions for the individual postural subscores of the REBA

This appendix contains the plots of the second degree polynomial regressions carried out in order to create a differentiable REBA. In each plot the red dots represent the data used as the base for the regression while the blue line is the plot of the found second degree polynomial.

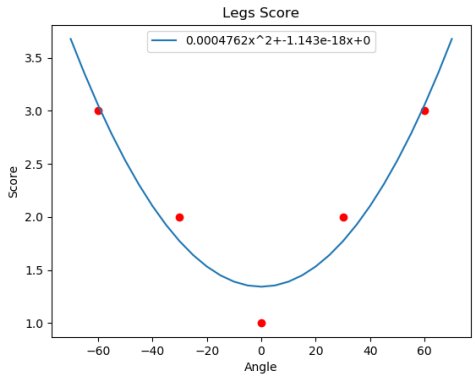


(a) The second degree polynomial found for the trunk score with the coefficient of determination of $R^2 = 0.935$.

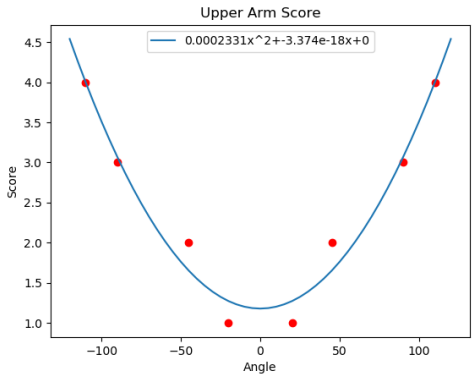


(b) The second degree polynomial found for the neck score with the coefficient of determination of $R^2 = 0.935$.

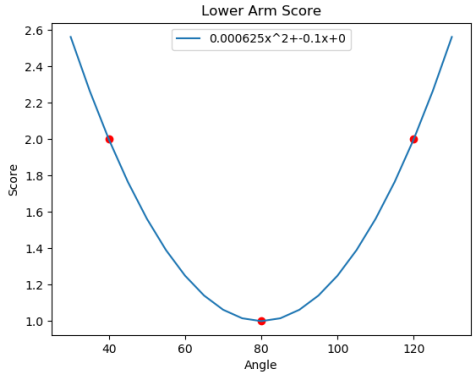
Figure C.1: Plots of the fitted second degree polynomial for the differentiable REBA.



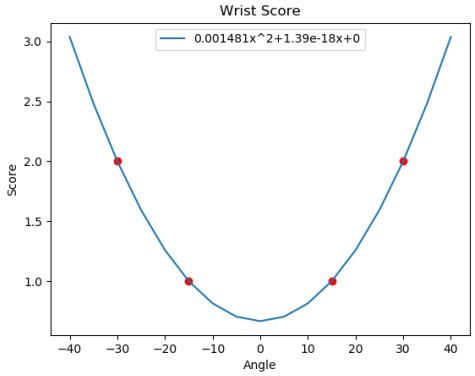
(c) The second degree polynomial found for the leg score with the coefficient of determination of $R^2 = 0.918$.



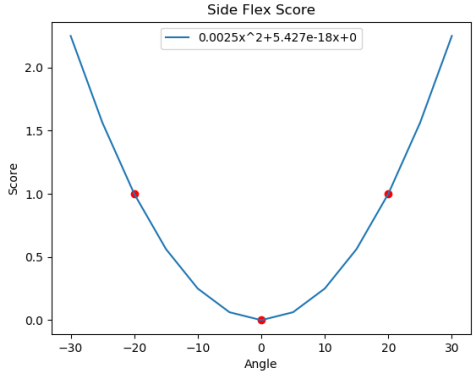
(d) The second degree polynomial found for the upper arm score with the coefficient of determination of $R^2 = 0.960$.



(e) The second degree polynomial found for the lower arm score with the coefficient of determination of $R^2 = 1.000$.



(f) The second degree polynomial found for the wrist score with the coefficient of determination of $R^2 = 1.000$.



(g) The second degree polynomial found for the side flexions, abductions and deviation scores with the coefficient of determination of $R^2 = 1.000$.

Figure C.1: Plots of the fitted second degree polynomial for the differentiable REBA (cont.).

Appendix D

Body Poses created and used to find the weights in the differentiable REBA.

Pose No.	Trunk	Trunk Side Flexion	Neck	Neck Side Flexion	Legs	Upper Arm	Abduction	Lower Arm	Wrist	Wrist Deviation	REBA Score
1	0	0	15	0	10	80	0	75	-10	0	1
2	0	0	5	0	25	-15	0	65	-15	0	1
3	0	0	20	0	10	30	20	50	15	0	2
4	-25	0	0	0	20	5	15	115	-15	0	2
5	0	0	5	0	70	10	0	80	10	10	3
6	0	0	10	0	10	60	20	110	20	0	3
7	30	20	10	0	0	60	20	30	10	0	4
8	-30	-10	15	0	50	-35	0	90	-20	0	4
9	-15	0	5	-15	75	95	0	70	-5	0	5
10	20	0	0	0	25	85	15	45	30	20	5
11	15	-10	10	15	75	-15	20	10	5	0	6
12	0	15	20	0	50	105	15	95	-10	-15	6
13	80	-10	-20	0	40	10	10	85	15	0	7
14	20	15	0	0	70	65	35	115	-5	10	7
15	-10	0	-10	20	40	100	20	10	0	0	8
16	70	-20	-20	-10	40	-40	0	20	10	-10	8
17	35	0	35	15	35	100	35	80	5	15	9
18	45	30	-15	-15	85	-35	10	50	-5	0	9
19	85	20	0	10	75	105	0	40	-30	0	10
20	70	-25	-10	0	55	95	30	25	-25	-20	10
21	70	10	-20	10	80	100	30	20	-20	0	11
22	70	20	-10	-20	70	120	30	120	25	15	11

Table D.1: The poses consisting of 10 angles in the body and the corresponding REBA score used for determining the weights. Angles are given in degrees (°).

Appendix E

Kinematic equations for the position of the contact point

This appendix provides the found kinematic equations for the x,y and z coordinate of the contact point at the right hand palm given the segment legnths and the body angles.

$$x = l_{\mathrm{pa}} \cdot ((((-\sin(\theta_{\mathrm{ts}}) \cdot \cos(\theta_{\mathrm{rua}}) \cdot \cos(\theta_{\mathrm{raa}}+\pi) + \sin(\theta_{\mathrm{raa}}+\pi) \cdot \cos(\theta_{\mathrm{ts}})) \cdot \cos(\theta_{\mathrm{rla}}) - \sin(\theta_{\mathrm{rla}}) \cdot \sin(\theta_{\mathrm{ts}}) \cdot \sin(\theta_{\mathrm{rua}})) \cdot \cos(\theta_{\mathrm{rwd}}) + (\sin(\theta_{\mathrm{ts}}) \cdot \sin(\theta_{\mathrm{raa}}+\pi) \cdot \cos(\theta_{\mathrm{rua}}) + \cos(\theta_{\mathrm{ts}}) \cdot \cos(\theta_{\mathrm{raa}}+\pi)) \cdot \sin(\theta_{\mathrm{rwd}})) \cdot \cos(\theta_{\mathrm{rw}}) - ((-\sin(\theta_{\mathrm{ts}}) \cdot \cos(\theta_{\mathrm{rua}}) \cdot \cos(\theta_{\mathrm{raa}}+\pi) + \sin(\theta_{\mathrm{raa}}+\pi) \cdot \cos(\theta_{\mathrm{ts}})) \cdot \sin(\theta_{\mathrm{rla}}) + \sin(\theta_{\mathrm{ts}}) \cdot \sin(\theta_{\mathrm{rua}}) \cdot \cos(\theta_{\mathrm{rla}})) \cdot \sin(\theta_{\mathrm{rw}})) + l_{\mathrm{la}} \cdot ((-\sin(\theta_{\mathrm{ts}}) \cdot \cos(\theta_{\mathrm{rua}}) \cdot \cos(\theta_{\mathrm{raa}}+\pi) + \sin(\theta_{\mathrm{raa}}+\pi) \cdot \cos(\theta_{\mathrm{ts}})) \cdot \cos(\theta_{\mathrm{rla}}) - \sin(\theta_{\mathrm{rla}}) \cdot \sin(\theta_{\mathrm{ts}}) \cdot \sin(\theta_{\mathrm{rua}})) + l_{\mathrm{pe}} - l_{\mathrm{sh}} \cdot \cos(\theta_{\mathrm{ts}}) - l_{\mathrm{sp}} \cdot \sin(\theta_{\mathrm{ts}}) + l_{\mathrm{ua}} \cdot (-\sin(\theta_{\mathrm{ts}}) \cdot \cos(\theta_{\mathrm{rua}}) \cdot \cos(\theta_{\mathrm{raa}}+\pi) + \sin(\theta_{\mathrm{raa}}+\pi) \cdot \cos(\theta_{\mathrm{ts}}))$$

$$y = l_{\mathrm{pa}} \cdot (((((-\sin(\theta_t) \cdot \sin(\theta_{\mathrm{rua}}) - \cos(\theta_t) \cdot \cos(\theta_{\mathrm{ts}}) \cdot \cos(\theta_{\mathrm{rua}})) \cdot \cos(\theta_{\mathrm{raa}}+\pi) - \sin(\theta_{\mathrm{ts}}) \cdot \sin(\theta_{\mathrm{raa}}+\pi) \cdot \cos(\theta_t)) \cdot \cos(\theta_{\mathrm{rla}}) - (-\sin(\theta_t) \cdot \cos(\theta_{\mathrm{rua}}) + \sin(\theta_{\mathrm{rua}}) \cdot \cos(\theta_t) \cdot \cos(\theta_{\mathrm{ts}})) \cdot \sin(\theta_{\mathrm{rla}})) \cdot \cos(\theta_{\mathrm{rwd}}) + (-(-\sin(\theta_t) \cdot \sin(\theta_{\mathrm{rua}}) - \cos(\theta_t) \cdot \cos(\theta_{\mathrm{ts}}) \cdot \cos(\theta_{\mathrm{rua}})) \cdot \sin(\theta_{\mathrm{raa}}+\pi) - \sin(\theta_{\mathrm{ts}}) \cdot \cos(\theta_t) \cdot \cos(\theta_{\mathrm{raa}}+\pi)) \cdot \sin(\theta_{\mathrm{rwd}})) \cdot \cos(\theta_{\mathrm{rw}}) - (((-\sin(\theta_t) \cdot \sin(\theta_{\mathrm{rua}}) - \cos(\theta_t) \cdot \cos(\theta_{\mathrm{ts}}) \cdot \cos(\theta_{\mathrm{rua}})) \cdot \cos(\theta_{\mathrm{raa}}+\pi) - \sin(\theta_{\mathrm{ts}}) \cdot \sin(\theta_{\mathrm{raa}}+\pi) \cdot \cos(\theta_t)) \cdot \sin(\theta_{\mathrm{rla}}) + (-\sin(\theta_t) \cdot \cos(\theta_{\mathrm{rua}}) + \sin(\theta_{\mathrm{rua}}) \cdot \cos(\theta_t) \cdot \cos(\theta_{\mathrm{ts}})) \cdot \cos(\theta_{\mathrm{rla}})) \cdot \sin(\theta_{\mathrm{rw}})) + l_{\mathrm{la}} \cdot (((-\sin(\theta_t) \cdot \sin(\theta_{\mathrm{rua}}) - \cos(\theta_t) \cdot \cos(\theta_{\mathrm{ts}}) \cdot \cos(\theta_{\mathrm{rua}})) \cdot \cos(\theta_{\mathrm{raa}}+\pi) - \sin(\theta_{\mathrm{ts}}) \cdot \sin(\theta_{\mathrm{raa}}+\pi) \cdot \cos(\theta_t)) \cdot \cos(\theta_{\mathrm{rla}}) - (-\sin(\theta_t) \cdot \cos(\theta_{\mathrm{rua}}) + \sin(\theta_{\mathrm{rua}}) \cdot \cos(\theta_t) \cdot \cos(\theta_{\mathrm{ts}})) \cdot \sin(\theta_{\mathrm{rla}})) + l_{\mathrm{sh}} \cdot \sin(\theta_{\mathrm{ts}}) \cdot \cos(\theta_t) - l_{\mathrm{sp}} \cdot \cos(\theta_t) \cdot \cos(\theta_{\mathrm{ts}}) + l_{\mathrm{ua}} \cdot ((-\sin(\theta_t) \cdot \sin(\theta_{\mathrm{rua}}) - \cos(\theta_t) \cdot \cos(\theta_{\mathrm{ts}}) \cdot \cos(\theta_{\mathrm{rua}})) \cdot \cos(\theta_{\mathrm{raa}}+\pi) - \sin(\theta_{\mathrm{ts}}) \cdot \sin(\theta_{\mathrm{raa}}+\pi) \cdot \cos(\theta_t)) - sqrt({l_{\mathrm{ll}}}^2 + {l_{\mathrm{ul}}}^2 - 2 \cdot l_{\mathrm{ll}} \cdot l_{\mathrm{ul}} \cdot \cos(\pi - \theta_{\mathrm{rk}}))$$

$$z = l_{\mathrm{pa}} \cdot (((((-\sin(\theta_t) \cdot \cos(\theta_{\mathrm{ts}}) \cdot \cos(\theta_{\mathrm{rua}}) + \sin(\theta_{\mathrm{rua}}) \cdot \cos(\theta_t)) \cdot \cos(\theta_{\mathrm{raa}}+\pi) - \sin(\theta_t) \cdot \sin(\theta_{\mathrm{ts}}) \cdot \sin(\theta_{\mathrm{raa}}+\pi)) \cdot \cos(\theta_{\mathrm{rla}}) - (\sin(\theta_t) \cdot \sin(\theta_{\mathrm{rua}}) \cdot \cos(\theta_{\mathrm{ts}}) + \cos(\theta_t) \cdot \cos(\theta_{\mathrm{rua}})) \cdot \sin(\theta_{\mathrm{rla}})) \cdot \cos(\theta_{\mathrm{rwd}}) + (-(-\sin(\theta_t) \cdot \cos(\theta_{\mathrm{ts}}) \cdot \cos(\theta_{\mathrm{rua}}) + \sin(\theta_{\mathrm{rua}}) \cdot \cos(\theta_t)) \cdot \sin(\theta_{\mathrm{raa}}+\pi) - \sin(\theta_t) \cdot \sin(\theta_{\mathrm{ts}}) \cdot \cos(\theta_{\mathrm{raa}}+\pi)) \cdot \sin(\theta_{\mathrm{rwd}})) \cdot \cos(\theta_{\mathrm{rw}}) - (((-\sin(\theta_t) \cdot \cos(\theta_{\mathrm{ts}}) \cdot \cos(\theta_{\mathrm{rua}}) + \sin(\theta_{\mathrm{rua}}) \cdot \cos(\theta_t)) \cdot \cos(\theta_{\mathrm{raa}}+\cos) - \sin(\theta_t) \cdot \sin(\theta_{\mathrm{ts}})\pi \sin(\theta_{\mathrm{raa}}+\pi)) \cdot \sin(\theta_{\mathrm{rla}}) + (\sin(\theta_t) \cdot \sin(\theta_{\mathrm{rua}}) \cdot \cos(\theta_{\mathrm{ts}}) + \cos(\theta_t) \cdot \cos(\theta_{\mathrm{rua}})) \cdot \cos(\theta_{\mathrm{rla}})) \cdot \sin(\theta_{\mathrm{rw}})) + l_{\mathrm{la}} \cdot (((-\sin(\theta_t) \cdot \cos(\theta_{\mathrm{ts}}) \cdot \cos(\theta_{\mathrm{rua}}) + \sin(\theta_{\mathrm{rua}}) \cdot \cos(\theta_t)) \cdot \cos(\theta_{\mathrm{raa}}+\pi) - \sin(\theta_t) \cdot \sin(\theta_{\mathrm{ts}}) \cdot \sin(\theta_{\mathrm{raa}}+\pi)) \cdot \cos(\theta_{\mathrm{rla}}) - (\sin(\theta_t) \cdot \sin(\theta_{\mathrm{rua}}) \cdot \cos(\theta_{\mathrm{ts}}) + \cos(\theta_t) \cdot \cos(\theta_{\mathrm{rua}})) \cdot \sin(\theta_{\mathrm{rla}})) + l_{\mathrm{sh}} \cdot \sin(\theta_t) \cdot \sin(\theta_{\mathrm{ts}}) - l_{\mathrm{sp}} \cdot \sin(\theta_t) \cdot \cos(\theta_{\mathrm{ts}}) + l_{\mathrm{ua}} \cdot ((-\sin(\theta_t) \cdot \cos(\theta_{\mathrm{ts}}) \cdot \cos(\theta_{\mathrm{rua}}) + \sin(\theta_{\mathrm{rua}}) \cdot \cos(\theta_t)) \cdot \cos(\theta_{\mathrm{raa}}+\pi) - \sin(\theta_t) \cdot \sin(\theta_{\mathrm{ts}}) \cdot \sin(\theta_{\mathrm{raa}}+\pi))$$

Where θ_{rk} is the right knee angle, θ_t is the trunk angle, θ_{ts} is trunk side angle, θ_{rua} is angle of the right upper arm, θ_{raa} is right arm abduction angle, θ_{rla} is the angle of the right lower arm or the elbow, θ_{rw} is the right wrist angle, θ_{rwd} is right wrist deviation angle, l_{sp} is the spine length, l_{sh} is the length from the spine to the shoulder joint, l_{ua} is the length of the upper arm,

l_{la} is the length of the lower arm, l_{pa} is the length of the palm of the hand, l_{pe} is the length from the spine to the hip joint, l_{ul} is the upper leg length and l_{ll} is the lo length.

Appendix F

The code for the variation method

The python code for the variation method proposed in section 5.8. The three functions `forward_kinematics_x()`, `forward_kinematics_y()` and `forward_kinematics_z()` are placeholders for the corresponding kinematic equation provided in Appendix E

```
import numpy as np
import random
import math

if __name__=="__main__":
    random.seed(42)
    optipose=[0.0,0.0,10.0,0.0,0.0,0.0,0.0,0.0,0.0,80.0,80.0,
        0.0,0.0,0.0,0.0]
    angle_num=len(optipose)

    ###########parameters##############
    #m_i
    vary_margins=15
    #s_i
    sampling_number=31 #should be an uneven integer
    #d
    req_min_distance=25
    #number of poses to find in this run
    number_of_poses=1000
    #k
    k_last=5
    ####################################

    #generate the sampling set
    larm_sample_array=np.linspace(optipose[10]-vary_margins,optipose
        [10]+vary_margins,sampling_number)
    neck_sample_array=np.linspace(optipose[2]-vary_margins,optipose
        [2]+vary_margins,sampling_number)
    genral_sample_array=np.linspace(optipose[0]-vary_margins,optipose
        [0]+vary_margins,sampling_number)
    knee_sample_array=np.linspace(optipose[0],optipose[0]+vary_margins
        ,sampling_number)

    found_poses=[optipose[:]]
    cont_mean_opti_diff=[[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]]

```

```
    for i in range(1,number_of_poses):
        temp=[]
        distance_condition=False

        while distance_condition==False:

            randomlist = []
            #generate a set of random indices
            for k in range(0,angle_num):
                randomlist.append(random.randint(0,sampling_number-1))

            #obtain the resulting sample using the set of random
               indices
            temp=[genral_sample_array[randomlist[0]],
               genral_sample_array[randomlist[1]],neck_sample_array[
               randomlist[2]],genral_sample_array[randomlist[3]],
               knee_sample_array[randomlist[4]],genral_sample_array[
               randomlist[5]],genral_sample_array[randomlist[6]],
               genral_sample_array[randomlist[7]],genral_sample_array[
               randomlist[8]],larm_sample_array[randomlist[9]],
               larm_sample_array[randomlist[10]],genral_sample_array[
               randomlist[11]],genral_sample_array[randomlist[12]],
               genral_sample_array[randomlist[13]],genral_sample_array
               [randomlist[14]]]

            distanc_checks=[]
            #check the distance to the k last postures
            for s in range(1,k_last+1):
                idx=i-s
                if idx >= 0:
                    distance=np.linalg.norm(np.array(temp)-np.array(
                       found_poses[idx]))
                    if distance>=req_min_distance:
                        distanc_checks.append(True)
                    else:
                        distanc_checks.append(False)
            if all(distanc_checks):
                distance_condition=True
            else:
                print("Discarded point due to insufficient distance")

        found_poses.append(temp)

    #find cartesian points
    cart_endpoints=[]
    sp, sh, ua, ca, pe, th,la,ha =
       [0.469,0.159,0.271,0.418,0.111,0.429,0.257,0.088]
    for pose in found_poses:
        ta, tsa, uaa, ab, ka, laa, de, wa = [pose[0]*(math.pi/180),
           pose[1]*(math.pi/180),pose[5]*(math.pi/180),pose[6]*(math.
           pi/180),pose[4]*(math.pi/180),pose[9]*(math.pi/180),pose
           [12]*(math.pi/180),pose[11]*(math.pi/180)]
        x=forward_kinematics_x()
        y=forward_kinematics_y()
        z=forward_kinematics_z()
        cart_endpoints.append([x,y,z])
```

Appendix G

Histograms of the Sampled Angles for the Joints

Plots showing the histograms of the sampled angles for each joint. Plots for 13 of the joints are provided here, while the plots for the last two joints (trunk and knees) can be found in section 5.9.

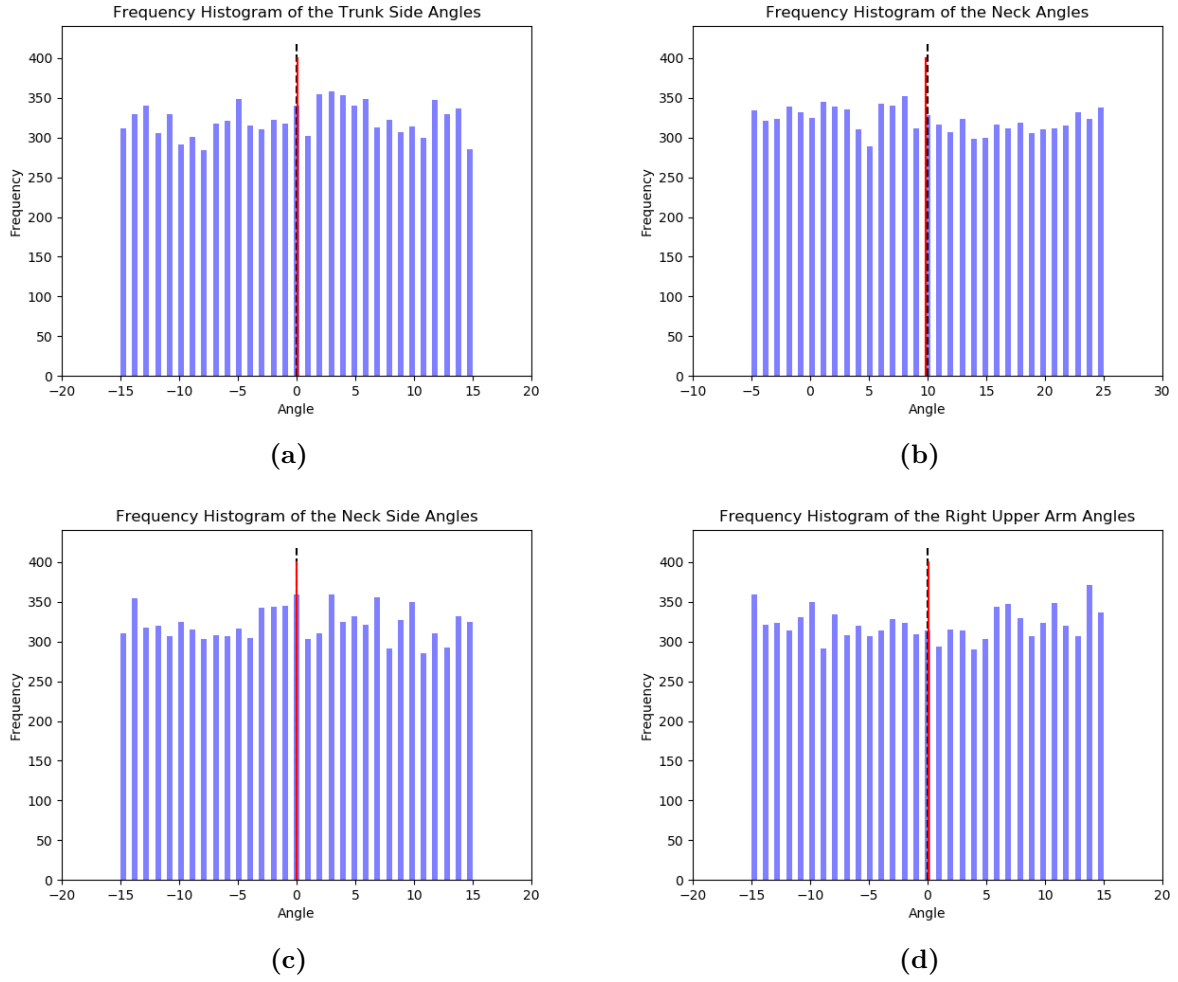
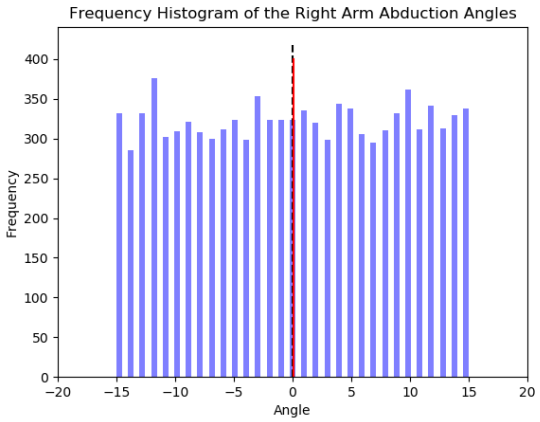
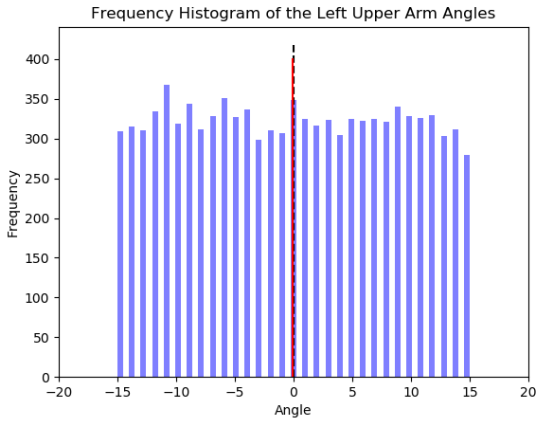


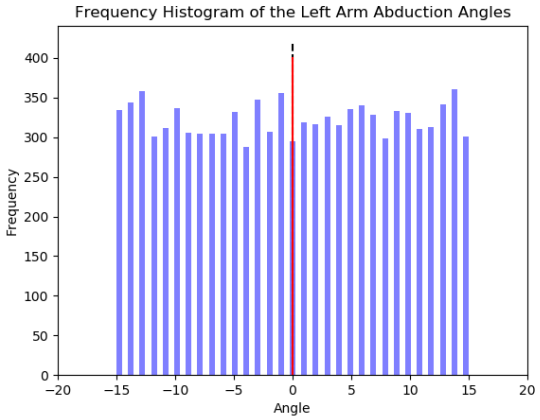
Figure G.1: The histograms of the 10,000 angles for each joint found using the variation method presented in section 5.8. The red line indicates the mean of the sampled angles. The dotted black line indicates the optimal angle.



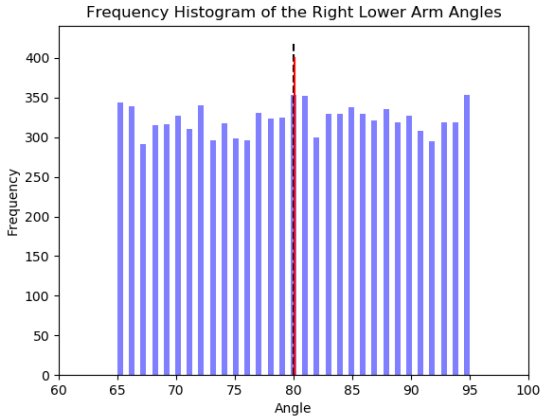
(e)



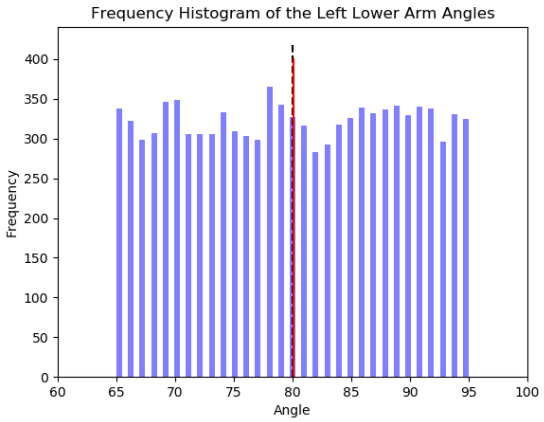
(f)



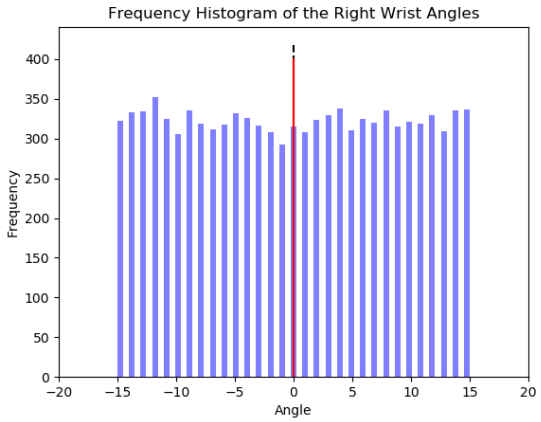
(g)



(h)

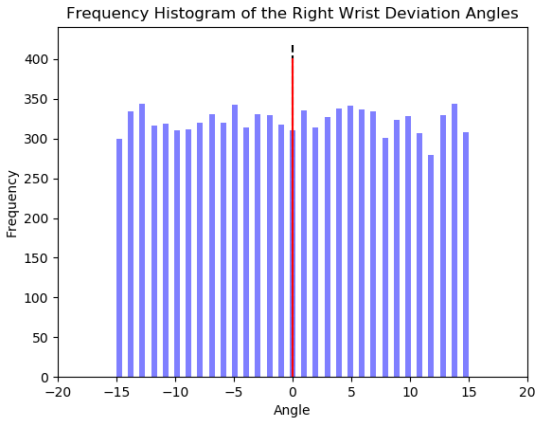


(i)

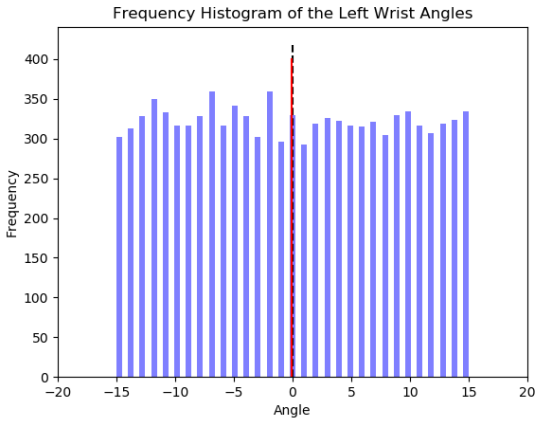


(j)

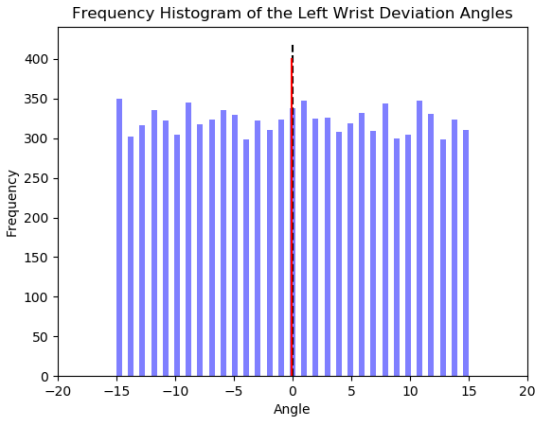
Figure G.1: The histograms of the 10,000 angles for each joint found using the variation method presented in section 5.8 (cont.). The red line indicates the mean of the sampled angles. The dotted black line indicates the optimal angle.



(k)



(l)



(m)

Figure G.1: The histograms of the 10,000 angles for each joint found using the variation method presented in section 5.8 (cont.). The red line indicates the mean of the sampled angles. The dotted black line indicates the optimal angle.

Appendix H

Optima-mean Difference Plots

Plots of the difference between the optimal angle and the sample mean as the number of samples increases. Plots for 13 of the joints are provided here, while the plots for the last two joints can be found in section 5.9.

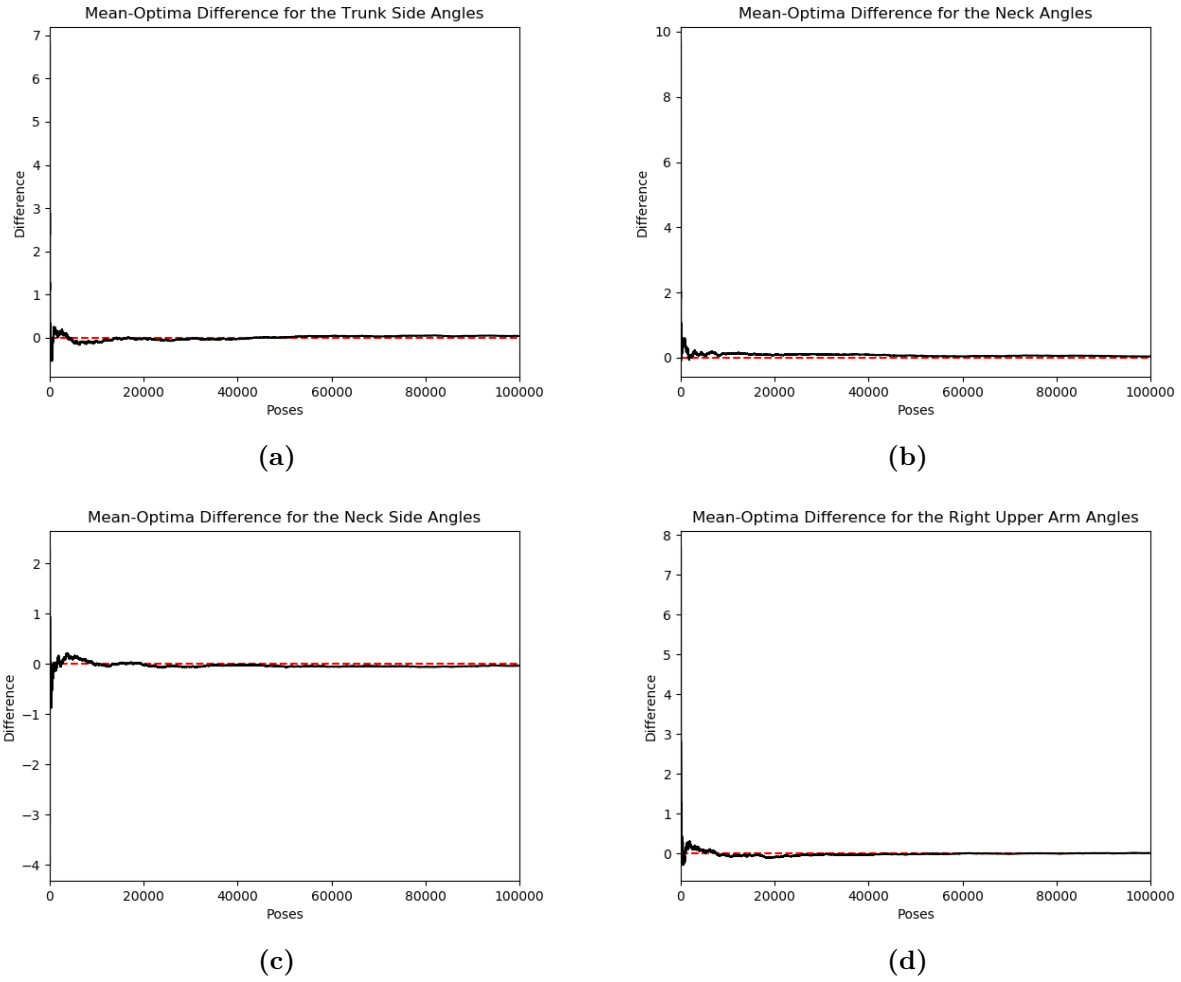
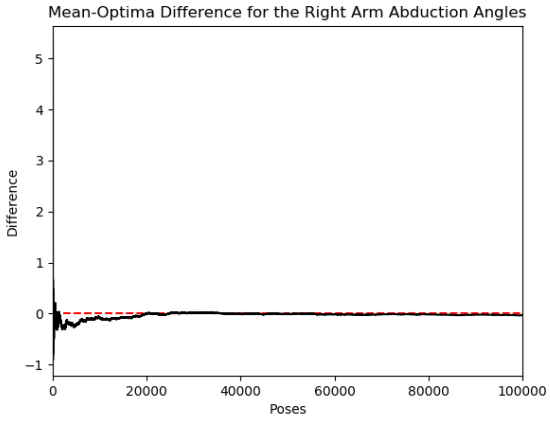
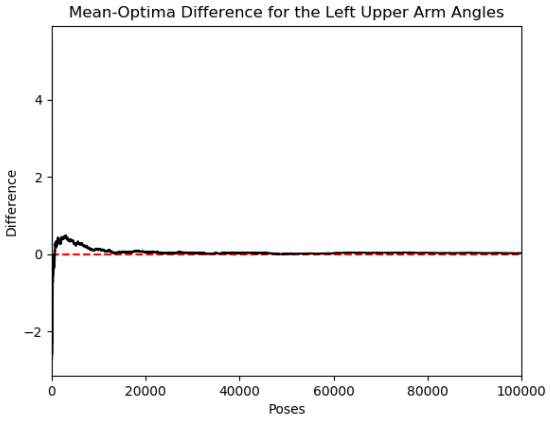


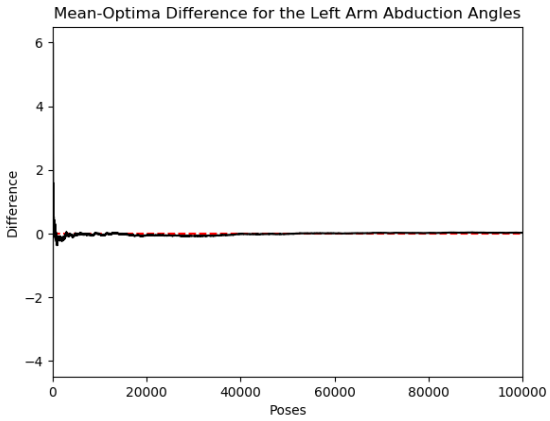
Figure H.1: Plots of the difference between the optimal angle and the sample mean as the number of samples increases. The red Line indicates the zero difference line.



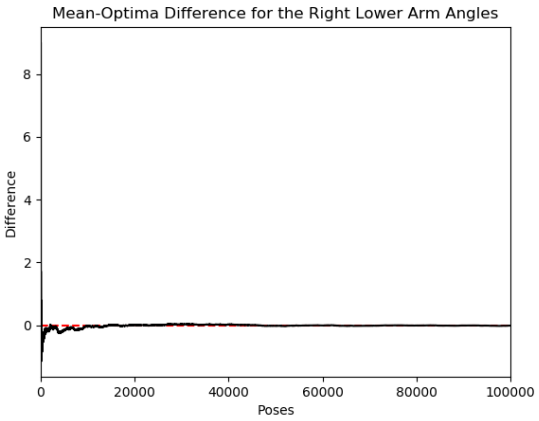
(e)



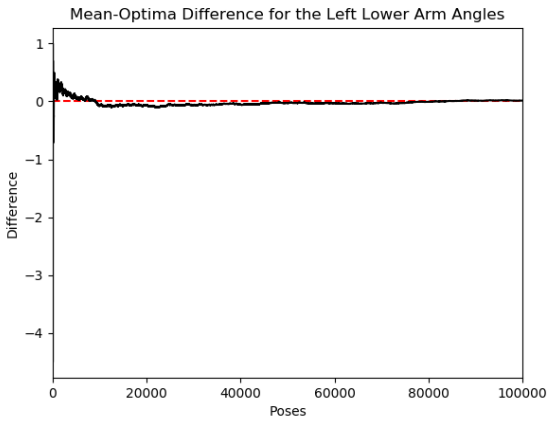
(f)



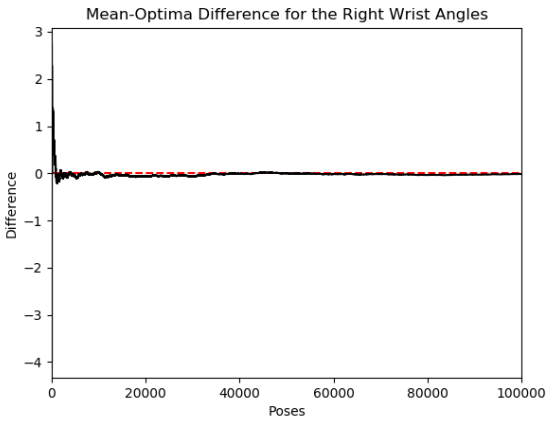
(g)



(h)

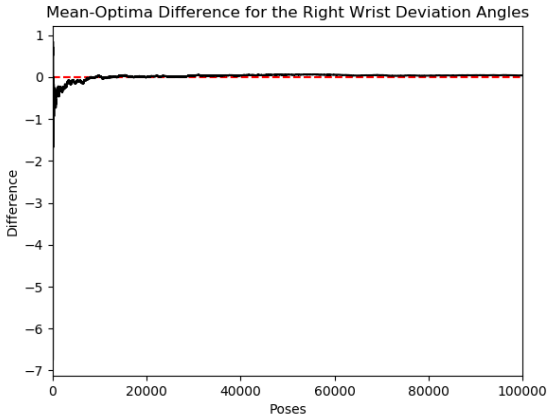


(i)

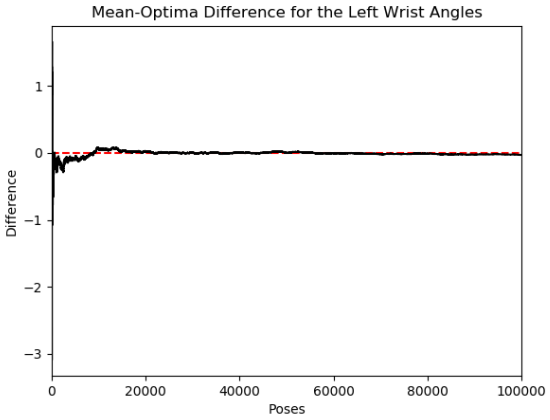


(j)

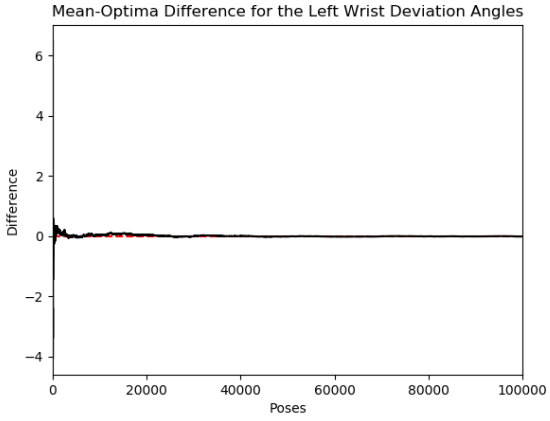
Figure H.1: Plots of the difference between the optimal angle and the sample mean as the number of samples increases (cont.). The red Line indicates the zero difference line.



(k)



(l)



(m)

Figure H.1: Plots of the difference between the optimal angle and the sample mean as the number of samples increases (cont.). The red Line indicates the zero difference line.