Aalborg University

Semester project

Real-Time Trust Assessment Based on Motion Data for Safe Human-Robot Collaboration

Written by:
Ozren Vučićević

Supervised by:
Dimitris Chrysostomou
Matthias Rehm

Department of Materials and Production
Department of Architecture, Design and Media Technology

Autumn semester 2022.

Dec 21th 2022

AALBORG UNIVERSITY
STUDENT REPORT

Aalborg University
Department of Materials and Production
Department of Architecture, Design and Media Technology
http://www.mp.aau.dk

Title:

Real-Time Trust Assessment Based on Motion Data for Safe Human-Robot Collaboration

Theme:

Trust assessment

Project period:

Autumn semester 2022.

Project group:

Participant:

Ozren Vučićević

Supervisors:

Dimitris Chrysostomou

Matthias Rehm

Number of pages: ??

Date of completion:

Dec 21th 2022

Abstract

This paper focuses on the trust relationship in close proximity collaboration of humans and robots. After experiments which were used to gather motion data, the data is sorted and analysed to be later used with the participants trust scores towards the robot to create a way to correlate the movement to the trust relationship. Different classification algorithms were used to correlate the trust with the movement data in hopes to, after further research, have a model which could measure trust in real time and increase the safety and productivity of the collaboration between humans and robots in an industrial setting.

Table of Contents

1 Introduction

As the field of robotics is developing, robotic systems are getting more automated and collaboration in close proximity with humans is increasing. Introduction of robots improved a lot of areas in our every day life. Science, engineering, medicine and a lot of other areas including the social aspects have rapidly improved with this collaboration and the connection between humans and robots will continue to grow. This development is bringing its own set of challenges and problems.

There is a vast range of applications regarding the robotic systems which include tasks such as package delivery, entertainment industry, disaster search and rescue, service, and manual support for physical tasks. Especially in engineering industry, which is the focus of this report, it is common to see humans collaborating with robots on a day-to-day basis, working together on tasks or in close proximity to each other. Many companies have big industrial robots which, if not careful, have the possibility to be dangerous to humans. This brings up the question: "Should humans trust robots?"

This is a complicated statement because, it is hard to look at machines as a agent that can or can not be trusted. Questions similar to this were a big topic of science-fiction and academia long before robotics got as advanced and developed like in today's society. In 1941, author Isaac Asimov wrote a science fiction novel "I, Robot", where he stated the Three Laws of Robotics [25]:

1. A robot may not injure a human being or, through inaction, allow a human being to come to harm.
2. A robot must obey the orders given it by human beings except where such orders would conflict with the First Law.
3. A robot must protect its own existence as long as such protection does not conflict with the First or Second Law.

Even if these laws are a part of science-fiction novel, they can provide an insight in the way engineers and researchers view the relationship of the human-robot collaboration. Safety is the most important factor when doing collaborative tasks and after that condition is satisfied, other aspects like productivity can be explored and improved. Human-robot interaction is not a new area of research, but only recently a lot of papers started being published.

In an effort to explore area of human-robot interaction, Aalborg University's HRI department is working on a European Commission project within Horizon 2020 Framework Program called "DrapeBot". The idea of this project will be to improve collaborative draping of carbon fiber parts. Draping carbon fiber on complex geometry is currently done by humans, so the robots will collaborate by helping to hold and carry the carbon plates, while the humans will drape them in to desired shape. [9]

1.1 Human-Robot Interaction (HRI)

Study between interactions of humans and robots is called human-robot interaction, also referred by researchers as HRI. This is a multidisciplinary field of science that has contributions from artificial intelligence, human-computer interaction, robotics, design, psychology and natural-language understanding. [24]

As a broad division HRI can be divided in four areas of application [20]:

1. Human supervisory control of robots in performance of routine tasks.
2. Remote control of robots in airborne, terrestrial , undersea and space vehicles for tasks in different hard-to-access environments.
3. Automated vehicles that have a human as a passenger.
4. Human–robot social interaction.

This report focuses on the 1. division. In an industrial environment there is a large variety of robots doing different assembly line task like welding, pick and place operations, painting etc. Humans working with industrial robots have supervisory tasks or they work in collaboration with them.

A good example is the Baxter assembly line robot, shown in Figure 1.1, which is a product made in Boston by Rethink Robotics. It is designed with intention on operating in close proximity with workers, while regarding their safety. It is designed to be mechanically compliant, like the human body. One of interesting innovations are the eyes of Baxter robot which do not allow robot to see, but have u purpose to communicate with the operator on what the robot is currently doing. Also, the arms of the robot are mechanically compliant, which enables the operator to move its hands and teach it manipulation task or to work in close proximity to one another.

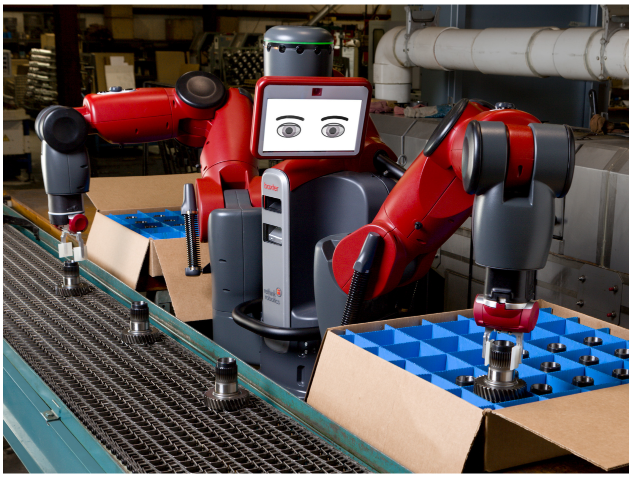


Figure 1.1: *Baxter robot*

Human–robot interaction (HRI) is a growing and rapidly expanding field which will require greater involvement of design and research. To date, human factor involvement in to human-robot research is limited and in the future it will be needed more then ever. As the human race is evolving at a slow pace, unlike robots, a need for specific conclusions about HRI will become invalid in the future.

1.1.1 Human-Robot Collaboration (HRC)

Human-robot collaboration (HRC) was briefly introduced in the last chapter on the example of the Baxter robot. Psychology, adaptation and learning process issues can happen in human-robot collaboration where the human needs to adapt to the industrial environment. Work-space that includes robots can cause a negative response by the humans, who can perceive them as dangerous and outrageous. [13] Human behaviour can be used as a source of information for the robot to produce a response in terms of reducing the cycle time and waiting time. [6] Parameters like movement, muscle activity, visual cues and brain signals can signalise human intention in the collaborative tasks, to improve the interaction. It is clear that in the real-world scenarios, understanding human intention and behaviour with cues like this will be a difficult problem to solve, because of the uncertainty and vagueness of humans.

As mentioned before, that negative response to a robot in the work-space can have an impact on the safety perception by the workers. This can influence the trust level, comfort, ability to predict and control the situation and experience of the worker. [6] As an example, as mentioned in [21], robot moving toward a human in higher speeds will increase the safety concerns by the human and he will find difficult to predict the robot's movements. This provides an insight in to the importance of the human emotional status when doing collaborative tasks with the robot. This psychological status is important for maintaining performance on an optimal level.

1.2 Trust

Trust, as referred by the Oxford English Dictionary is a "firm belief in the reliability, truth, or ability of someone or something" or the "confident expectation of something." COBUILD Advanced English Dictionary by Collins also refers to a "firm belief or confidence in the honesty, integrity, reliability, justice, etc. of another person or a thing."

In common usage, trust is the expectation that something good will happen, while also having the knowledge that it might not. It is mostly considered a human concept, because in most cases only a person can decide what constitutes something trustworthy or not. There are different ideas of trust and we can divide them in to the idea of "Performance trust" and "Moral trust". Performance trust is referring to the belief that the trustee is capable of task completion, while moral trust refers to the trustor's confidence that the trustee is choosing the right action morally. When referring to robots, performance trust is the first idea that comes to mind as something that can be measurable.

This paper focuses on the human-robot trust relationship. [5]

1.2.1 Human-Robot Trust Relationship

The work in this field is focused on the reliability and ability characteristics of robotic systems. One of the examples of that is a meta analysis of factors that prior research identified as the most relevant points that dictate trust in human-robot interactions. [17]

After collecting data from 21 studies, the researches found out that factors related to the robot—specifically, its performance and consistency, were the biggest association with trust. Applying human factors (e.g., comfort, attitudes, happiness) and environmental factors had the smaller contribution. A person can gain or lose trust in the robot, without looking the robot as a moral agent.

The problem with human-robot trust comes when you try to measure it. When looking to human-human relationships trust is often signified by one or more moral characteristics (e.g., benevolence, integrity, sincerity) and performance characteristics (reliability, ability). Regarding that point, a distinction has been made in [22] to separate trust into "relation-based" trust and "performance-based" trust. Performance-based trust revolves around the idea that the robot is trusted to be capable, competent and reliable to do its task, without the need of supervision or monitoring. Performance-based trust can also depend on the robot's predictability, responsiveness and transparency. On the other hand, relation-based trust revolves on the idea of participant trusting the robot to be a part of the society and not as a part of the factory doing a job, with no regard or knowledge of social norms. When comparing these two, relation-based trust is given to robots who would function good in human-robot interaction, for example as children toys and parts of the every day life, and performance-based trust would be given to factory robots who are doing a precise task. Usually, robots that have performance-based trust are barred of from human interaction. Currently there is not a lot of ways to measure trust between humans and robots, but there are some standardised questionnaires that can be used as trust indicators. One of them was used in this report.

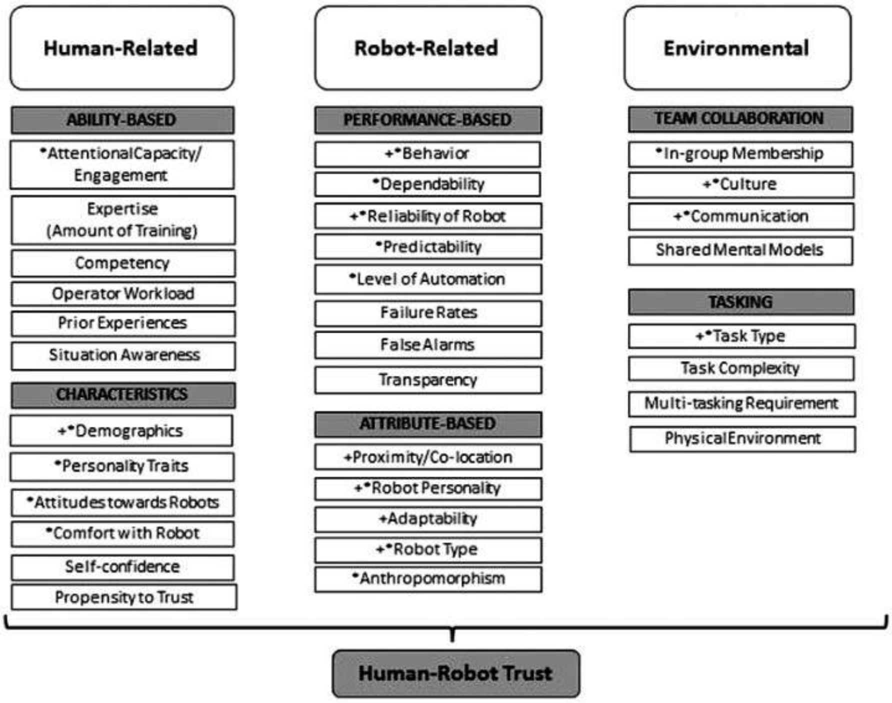


Figure 1.2: *Factors of trust development in human-robot interaction [17]*

1.2.2 Objective Measure of Trust

In experiments, the primary means of measuring the participant's trust in a robot are the subjective questionnaires. Lot of studies use this subjective questionnaires and rely on them to validate their trust measurements, others create their own. There are validated questionnaires and non-validated ones, which are usually made by the study itself. In general, this subjective measurement methods are used more often then objective ones. Objective measures allow the researchers to analyze actually robot interaction, instead of relying on subjects speculation about themselves (e.g., reason, motivation, beliefs, etc.) Since trust in automation was used mainly for evaluation purposes, most of the methods for trust assessment are based on the subjective questionnaires mentioned before. Also, when there is a need to measure trust in real time, it is not practical to have constant trust questions answered by the humans. Because of this, new methods of trust estimation are needed to be tested. Unlike subjective psychological indicators of trust, there is a possibility that trust can also be measured by some physical indicators. Some of those indicators can be human movement or hearth rate. This report is exploring the possibility of trust indicators in humans based on their movement. To achieve that, a study called Laban Movement Analysis is used to extract the different movement descriptors, which can be used to further analyse the behaviours based on the movement data. [3]

1.3 Machine Learning With Human Movement Data

As the machine learning and artificial intelligence (AI) algorithms are rapidly improving, they are being used in a lot of different sectors, private and public, to make basic and complex decision-making processes more simple. Computers are a good way to analyse and work with large data-sets. They have simplified extracting insights from data that would make less sense for a human to analyse. In toady's society, algorithms like this are using large amounts of data to influence peoples lives in a variety of areas, from adds on the internet or assisting banks with creditworthiness of its members. [16]

In supervised machine learning, algorithms are trained with "labelled" training data, and on basis of that data they will predict the output. The labelled data is input data that is already tagged with a correct output. In this situation, this training data that is placed in to the algorithm works as the supervisor, that will teach the machine to predict the correct output. We can relate that to a student that learns with the supervision of the teacher. Often this algorithms depend on large amount of data sets, or training data. [15]

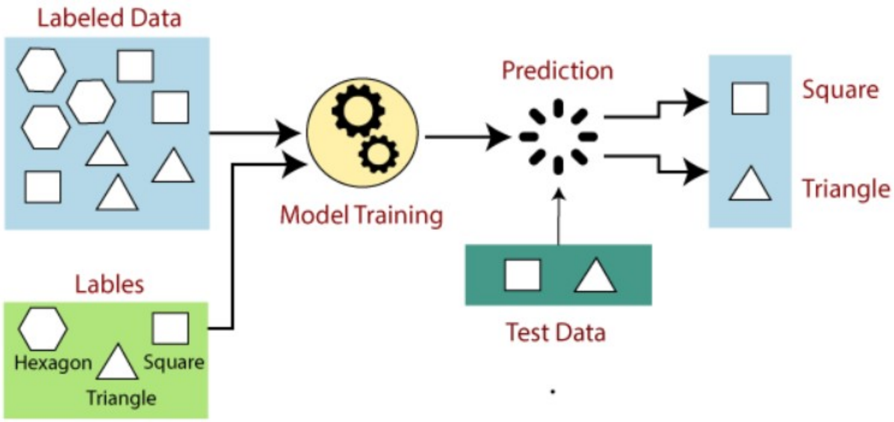


Figure 1.3: *Supervised machine learning [12]*

There are several ways to record human movement. The usual methods are using cameras or motion-tracking devices which are placed on the body of the test subject. Both methods usually create big data-sets for further analysis. A good example of research like this is a study, shown in [26], about the motion data analysis by supervised machine learning algorithms. This research was intended to developer technology which will assist robots to adapt to human behaviour patterns in real-time to maximize the system efficiency, understand human behaviour and movement, and show effects in ergonomics and assist robotics. Motion trackers were placed on the participants to gather motion-data, which was used to train classification machine learning models. As explained in the article, this kind of approach to tasks related to analysis of the human movement and behaviour is most common, because a large amounts of labeled examples are created by recording movement.

2 Problem Analysis

In today's industry, big industrial robots that can cause harm to humans are kept in cages with safeguards so injuries can be avoided. In the future, the hope is that human operators will trust the robots enough to work with them in close proximity on collaborative tasks, while also providing the safety in their interaction by the robots. Trust between humans and robots, as mentioned in Section 1.2.1, affects productivity and speed of finishing tasks, so in collaboration having a good model for indication of trust can be beneficial for overall performance.

To enable close-proximity human-robot collaboration (HRC), part of the focus has to be put on the trust relationship between the human and the robot. In this paper that trust relationship is explored and measured to approximate the appropriate levels of trust in the operator toward the robot, because that level will influence the behaviour of the operator in terms of performance and proximity to the robot. As mentioned in Section 1.2.1, a division has been made in two forms of trust, relation-based trust and performance-based trust. Relation-based trust is becoming increasingly important in toady's society and it will be important to understand it better, but this paper will focus more on the performance aspects of trust.

As mentioned in Section 1, Aalborg University is working on a European Commission project. To replicate the environment of collaborative draping of carbon fiber, a series of experiments has been conducted by Aalborg University's HRI department, with a purpose to gather movement data which will be analysed.

2.1 Problem Description

The "DrapeBot" project [9] revolves around collaborative draping of carbon-fiber between humans and robots. Draping of composites is a procedure where thin layers of composite material, in this case woven carbon-fiber, are stacked on top of each other to fit a chosen geometrical structure. When possible, this procedure can be automated and no human involvement is needed. As explained in [1], for more complicated shapes, this process is often done manually by workers. Because of that, the price of production increases with additional labour costs. To increase the productivity and speed, a collaborative work that includes a robot and a worker would fit this scenario. Also, when working with composite plates that are heavy and hard to handle, a robot will have to transport them in the facility. The whole process would include the robot, who would carry and hold the carbon-fiber sheets, and a worker that would drape them around a specific geometry.

Figure 2.1: *Draping of composites*

Aalborg University's HRI department conducted a series of experiments to emulate the carbon-fiber draping and gather data. Data was gathered by the Xsens MVN Awubda tracking suit and trust questioners from [19], which will be used to test the hypothesis of the project. The hypothesis focuses on the idea that the performance-based measurements of the humans trust towards the robot and human movement patterns are able to be a good indicator of trust levels towards the robot, which will be used for identifying critical points of the human-robot interaction. The belief is that some of the unintentional body movements and patterns of movement of the human in the interaction can be used to approximate the trust level and determine when the trust level is inefficient or dangerous. Trust level can ether be to high which is dangerous. In this case, that would mean that the human would be too comfortable working with the robot and endangered his safety by standing in to close proximity. If the trust level would be inefficient, that would mean that the human would not trust the robot, so as mentioned in Section 1.2.1, performance and efficiency of completing the collaborative task would be lowered. It is also important to note, that the trust relationship will be changing as the interaction is happening. The trust levels at the beginning of the experiments will evolve depending on the different parameters of the interaction. That is why, it is important to have the data segmented in time, in relation to the task that is happening. In the future, this trust relationship would be evaluated in real time and the robot actions would be corrected appropriately.

Figure 2.2 shows how assessment of trust with trust calculations and motion data can be used to calibrate trust, by communicating the operator or by changing the robot behaviour.

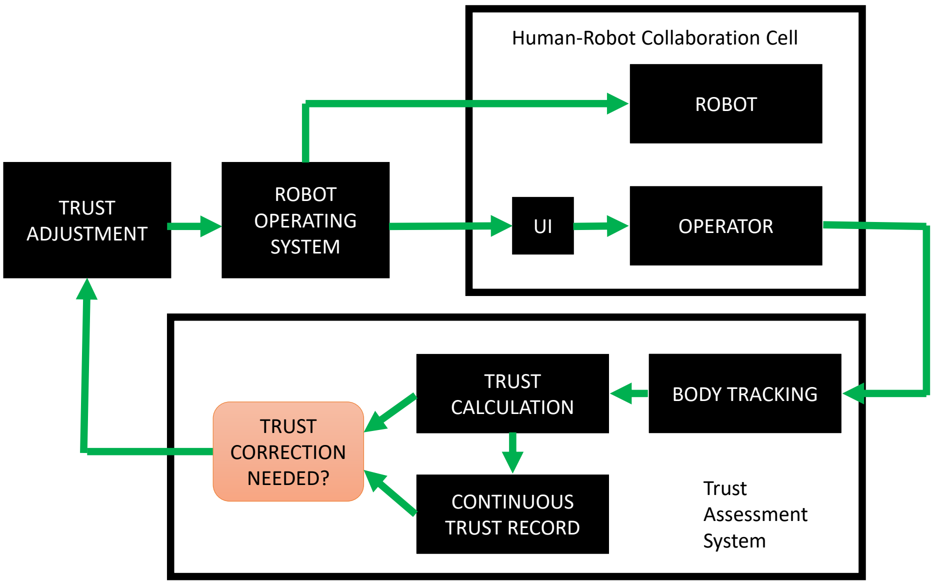


Figure 2.2: *Trust assessment system involving a human-robot collaboration cell*

In the "DrapeBot" HORIZON2020 project Human-Robot collaboration cell would contain a worker and robot doing a collaborative close-proximity draping task, in witch motion data, or different body data, from the worker would be collected, with which the trust calculation would be conducted. If there would be a need for trust adjustment, either the robot would change its pattern of behaviour or the operator would be notified by the user interface that he needs to correct his behaviour. This would result in a safer, more productive and efficient system of close-proximity collaboration of draping between the worker and the robot.

2.2 Problem Statement

The purpose of the HORIZON2020 project is to investigate the close-proximity collaboration of humans with robots during the task of collaborative draping of carbon-fiber plates. The goal is to use trust of the human towards the robot, which will dictate the work relationship between them. The trust of the worker can not be too high or too low, because that has a influence in safety and performance of the close-proximity collaboration.

In respect to the European project, this paper shares it's goals, but since the project is in its early stages only limited amount of data and research has been done. Problem statement of the project is:

A) How can the experiment movement data gathered be separated and processed?

B) How does the trust of the participants towards the robot correlate to the movement data?

C) Can the trust analysed by the movement data be used as a tool in collaborative close-proximity human-robot tasks?

To get answers to the problem statements, a workflow of the project followed this sequence:

1. Analysis of the experiment data gathered by the HRI department of Aalborg University
2. Movement data separation based on relevant factors and feature calculation with the use of Laban Movement Analysis
3. Testing the gathered features with Supervised machine learning algorithms
4. Discussion on the gathered results

3 Experiment

One of the main ideas of this project is carbon-fiber draping. Draping of composites is a procedure where thin layers of composite material, in this case woven carbon-fiber, are stacked on top of each other to fit a chosen geometrical structure. As explained in [1], for more complicated shapes, this process is often done manually by workers. This can also be seen on Figure 2.1. Because of that, the price of production increases with additional labour costs. To increase the productivity and speed, a collaborative work that includes a robot and a worker would fit this scenario. The robot would bring the carbon-fiber sheets, while the worker would drape them.

To emulate this process AAU HRI department made a series of experiments as mentioned in Section 2.1. When trying to replicate the HORIZON2020 project idea, there were requirements that needed to be accomplished by the experiments. HRI department decided to emulate the collaborative task of transferring the carbon plates from one place to another and the task of draping in separate experiments.

For gathering the data, 20 participants were involved in two experiments, as shown in the Table 3.1:

A) Transport task (20)

B) Draping task (20)

Table 3.1 shows participants, with numbers from 1-20. Both experiments contained a big industrial robotic arm (KUKA 300 R2500), shown in Figure 3.1 and one test subject per experiment. Good thing about using a robot this size is the possibility of participants having less trust in the robot trough the encounter, even if the experiment was safe and no harm would happen to participants. This makes the experiment more realistic. Instead of carbon-fiber plates, robot and participants were carrying and draping textile material.

Figure 3.1: *KUKA 300 R2500*

3.1 Transport Task

Experiment A was the task of transport. The goal was to replicate the close-proximity task of transporting the carbon-fiber plates from one point to the other. As mentioned in Section 3, textile material was used in the experiment. One part of textile was connected to the robot arm's gripper, and the other was handled by the participant. In the experiment the textile was transferred from one table to another, while the participant was walking alongside the robot between the tables. The distance between them was the length of the textile sheet.

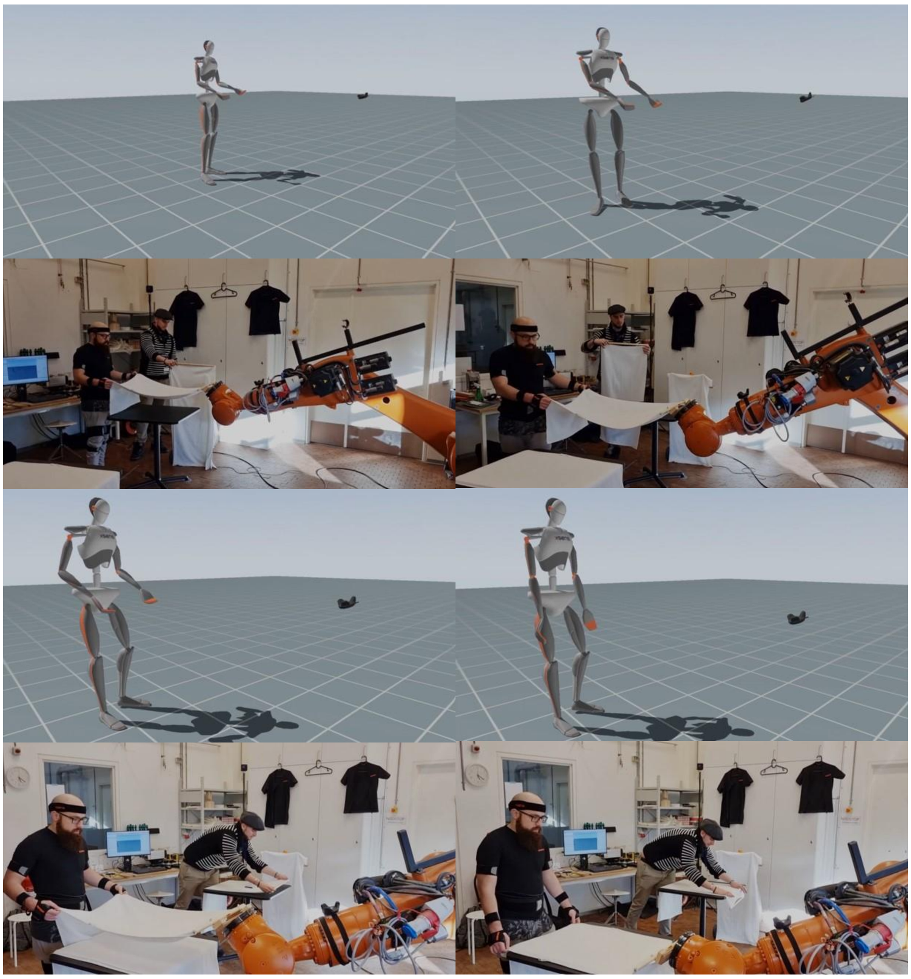


Figure 3.2: *Collaborative transport task*

Figure 3.2 shows the experiment conducted, with the images from the motion tracking program used.

3.2 Draping Task

Experiment B was the task of draping. The goal was to replicate the close-proximity task of draping the carbon-fiber plates over the complex geometry by the worker. To emulate this the textile material used in the experiment is placed on the table by the robot, while the participant waits. After the robot places the material, the participant approaches the table and drapes the textile material over the table in close proximity to the robot, and backs away. After that the robot returns to its original position and the task is repeated. When applied in the industrial context, this requires that the worker is comfortable with getting in to close proximity to the robot when it is stationary in order to avoid lost efficiency and stress. Images from the experiment and from the motion tracking software can be seen in Figure 3.3.

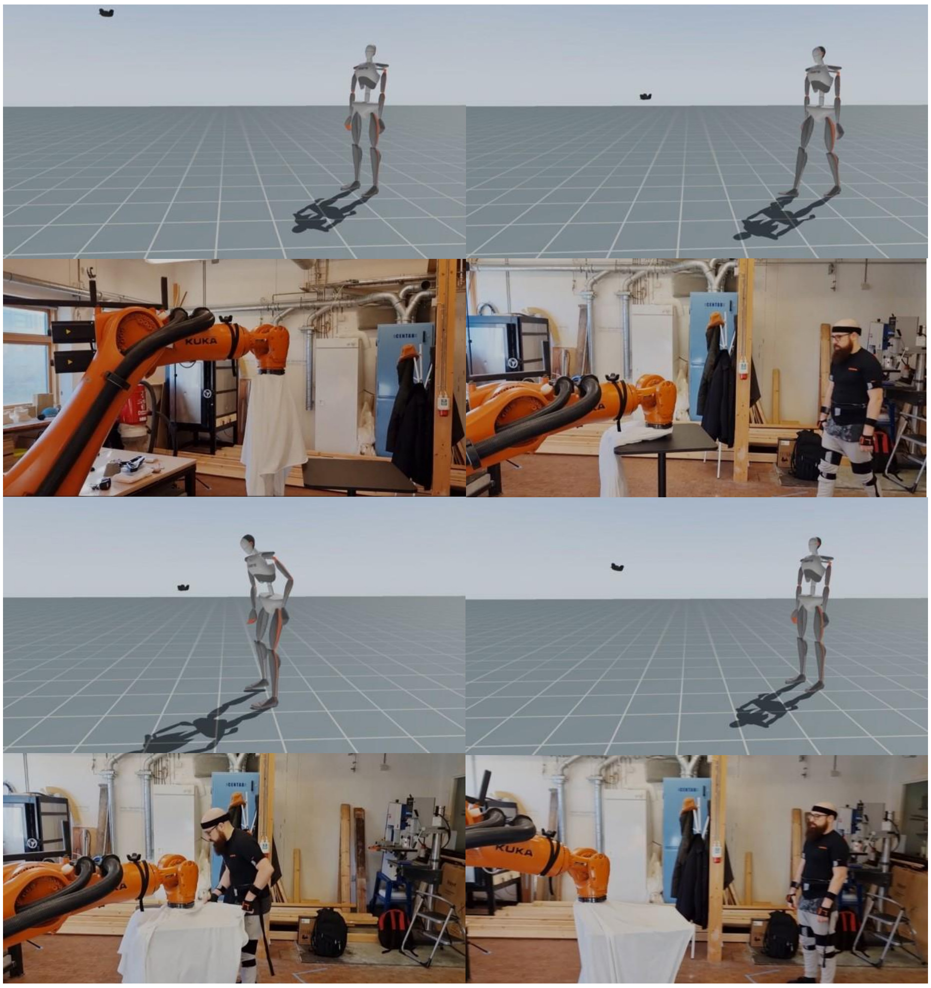


Figure 3.3: *Collaborative draping task*

3.3 Data Collection Procedure

The idea of data collection was to collect body movement data of the robot and the participants with the trust data collected from the questioners. Since this is a preliminary experiment, the only thing with the robot movement that was varying trough the experiment was the speed of movement, which can be seen in Table 3.1. Some participants would preform tasks with the higher robot speeds, and some with lower. Every task experiment lasted for approximately 10 minutes, in which participants would repeat the same task multiple times. After finishing the draping or transport task, participants would fill out a 14-question questionnaire regarding the trust in the robot, to provide a crude trust estimation. This means that every participants did 3 trust questionnaires, one before the first task, after the first task and the last after the second task.

Subject	First Task	Transport Speed	Age	Gender	DominantHand	Height
1	Transport	Slow	20	Male	Left	1.78
2	Draping	Slow	23	Male	Right	1.82
3	Transport	Fast	20	Male	Right	1.9
4	Draping	Fast	27	Male	Right	1.89
5	Transport	Slow	29	Male	Right	1.65
6	Draping	Slow	24	Male	Right	1.7
7	Transport	Fast	24	Female	Right	1.64
8	Draping	Fast	21	Female	Right	1.61
9	Transport	Slow	25	Male	Right	1.85
10	Draping	Slow	22	Female	Right	1.67
11	Transport	Fast	22	Female	Left	1.53
12	Draping	Fast	26	Male	Right	1.86
13	Transport	Slow	36	Female	Right	1.72
14	Draping	Slow	21	Male	Right	1.87
15	Transport	Fast	24	Male	Right	1.72
16	Draping	Fast	25	Female	Left	1.58
17	Transport	Slow	30	Female	Right	1.59
18	Draping	Slow	26	Male	Right	1.82
19	Transport	Fast	32	Male	Right	1.82
20	Draping	Fast	26	Male	Right	1.7

Table 3.1: *Experiment participants*

During the experiment, the participants were always at maximum distance from the robot, so there was no risk of injury, but it is presumed that the participants without prior robotic knowledge do not know the safety aspects of the experiment. In reality it is hard to replicate a real work environment between a worker and a industrial robot, so creating experiments like this are best ways of obtaining all of the needed data until testing on the real "DrapeBot" collaboration cell will be available.

3.4 Motion Tracking

Figure 3.4: *XSens MVN suit has 17 inertial and magnetic sensor modules. It is wireless so data is transmitted to the computer where the processing is visualised and performed*

For all of the experiments, motion tracking software was done using the XSens MVN Awinda tracking suit, shown in Figure 3.4. This is a wireless full-body motion capture system which consists of a tight-fitting shirt, gloves, a headband and a series of straps, which are used to attach 17 IMUs to the participant. An Inertial Measurement Unit (IMU) is a device that can measure and report specific gravity and angular rate of an object to which it is attached [23]. An IMU usually consists of:

- Gyroscopes: they provide a measure of angular rate
- Accelerometers: they provide a measure of specific force/acceleration
- Magnetometers: they provide a measurement of the magnetic field surrounding the system

For result accuracy, the participants body dimensions were measured before the tracking session. This measurements included height, foot length, shoulder height, shoulder width, elbow span, wrist span, arm span, hip height, hip width, knee height and ankle height. This information is needed for the system to separate the body in to different segments and joints. The XSens MVN Awinda system uses inverse kinematics to track the movements at a constant rate of 60 Hz, between the segments and joints of the body. The body segments and joints are not all connected to a specific sensor, but are rather calculated with the software and body dimensions. For this project it is important to note that there are 23 body segments. Segments are head, neck, four vertebra, left and right shoulders, upper arms, forearms, hands, upper legs, lower legs, feet, toes and the pelvis.

3.5 Trust Assessment

As mentioned in 1.2.2, in this types of experiments the most common way to get a trust measurement is with a trust questionnaire. In this experiment, a post-interaction questionnaire like that was also used. It was developed by Kristin Schaefer, as seen in [19], where the final trust score is measured based on a series of questions regarding the participants view of the interaction between the robot. Questions were meant to measure the robot's predictability, capability, error rates and more. Even tough this questionnaire consists of 40 questions, there is a 14 question version that was made for environments where the results need to be found quickly between repetitive tasks. The trade of that is a loss in granularity of the results. The final trust scores are measured on a scale from 0 to 100, where the participant's agreements to statements suggest trust in the robot and get a positive scale and statements that suggest error expectations get a negative score.

4 Methods

To analyse the data, different methods will be used in the report and described in this chapter. Firstly, methods that were needed to describe the captured human motion data are explained and afterwards methods for correlating the data with the trust scores are explained.

As a starting point, before any segmentation based on proposed methods was done, the first crude movement separation was done by analysis of the repetitive movement by treating it as a signal. This will describe movement in relation to the task that is happening in the experiment with segments that were discussed and agreed upon after the experiment. This will simplify the classification later on in the project.

After that, it was important to capture human movement patterns or signals that could indicate a correlation with the overall trust of the participant towards the robot. The experiment captured movement of all parts of the human body. This was possible with the Xsense MVN Awinda as mentioned in Section 3.4. For the purpose of defining the captured movement data, Laban Movement Analysis was used.

During the analysis of the movement, a Principal Component Analysis (PCA) was used as a dimensionality reduction tool.

After the movement data was analysed and defined by the movement analysis, the idea was to find correlations between the movement and the trust of the participant towards the robot. For analysing big data sets, as was required in this project, machine learning classification algorithms are a good way to do that, as mentioned in Section 1.3.

4.1 Related Work in Movement Analysis

When looking into movement segmentation methods, many are based on changes in the low-level kinematic features. For example [7] segments the movement in two different ways that utilize angular velocity of 4 different degrees of freedom in the arm. [27] is calculating boundaries where the hand linear acceleration and curvature are above some threshold. As can be seen, this papers are using low-level features, like acceleration, velocity and curvature of the body segments as markers for segmentation of movement. These kinematic methods are extremely efficient, but they produce simple segmentation.

There are more sophisticated motion capture segmentation methods, which produce a more high-level segmentation then the kinematics methods mentioned. They use data analysis with time series data by creating time clusters related to different motions. One of examples is shown in [2], where two different segmentation methods are implemented based on data compression. First method segments the data based on Principal Component Analysis (PCA), where the projection error increases on incrementally larger segments of motion capture data. The second method segments the data by spotting the changes in the distance of fitting a small segments of motion data to a Gaussian distribution model of the frames that precede the segment.

Some work has also introduced Supervised machine learning as a way of segmenting the data. As an example [14] uses a Bayesian classifier as a mean to derive choreographer segmentation profiles regarding the dance motion capture sequences. It is important to note, that to use methods like this on a general motion examples is extremely difficult and the amounts of data needed for general Supervised learning model of movement would be enormous. However, when using it for specific movements like dance motion it can be useful.

4.2 Laban Movement Analysis

Human motion is a field of study that is becoming increasingly active and includes motion analysis, motion recognition and motion synthesis. Capturing human motion and analysing it can be beneficial for a wide variety of applications. Most notable ones are film and game industry, human-computer interaction and ergonomics. Recently data related to motion capture is becoming free and available, which helps the study of movement. The problem with this data is that it rarely portrays the expressiveness of the human, which can indicate their state of mind, intent and emotion.

Laban movement analysis (LMA), also referred as Laban/Bartenieff movement analysis, is a method and language used for interpreting, describing, visualizing and documenting human movement. LMA has its roots in the work of Rudolf Laban, which was later extended by Irmgard Bartenieff, Lisa Ullmann and others. [3] It is based on different fields of research, like kinesiology, psychology and anatomy, and it has a vast area of usage. It is used by actors, musicians, dancers, health professionals, psychotherapists and others.

Other learning based methods of human motion segmentation have the ability to separate motion semantically. This is useful for specific class of motions, like dance moves or sign language, but when looking into general motion capture data, they wont be feasible. [4] When looking for a classifier for general motion, it has to have classes for all motions, which are significant in meaning and practical to use for classification. That is where LMA excels.

LMA is used to identify components that can describe the structural, geometric and dynamic properties of human motion. This analysis has four components:

- Body
- Space
- Shape
- Effort

Body, Shape, and Space are used to define what type of motion is performed, while the Effort describes how a motion is performed. Body and Space are used to describe how the human moves withing the 3D space or in relation with the body. The Effort relates to the intention of the movement and focuses on aspects of movement like dynamics and energy. Effort is composed of four different values.ž

Next two sections will explain what are the movement descriptors of the Laban Movement Analysis, that can be seen in the Table 4.1. Descriptors from this table, as well as the low-level and high-level division of the descriptors is proposed by [3].

Low-Level	High-Level
Kinematic / Dynamic Descriptors	Body Descriptors
Duration *Velocity* *Acceleration* *Jerk* *Curvature* *Quantity of movement*	*Action presence* *Center of mass displacement* *Balance* *Support*
Geometric Descriptors	Space Descriptors
Bounding box *Bounding sphere* *Bounding ellipsoid* *Convex hull* *Displacement* *Rotation* *Center of mass*	*Distance covered* *Area covered* *Hip height*
Effort Descriptors	Shape Descriptors
Weight effort *Time effort* *Space effort* *Flow effort*	*Bounding Volume* *Shape directional* *Shaping* *Extensiveness* *Arm shape* *Elbow shape* *Shoulder angle* *Hands relationship* *Feet relationship*

Table 4.1: *Laban Movement Analysis movement descriptors.*

4.2.1 Motion Representation

We can describe movement as a sequence of joint configurations of the skeleton in a selected period of time. For this project, each joint configuration at the time t_i can be described by:

- A set of m joint/segment positions: $\mathbf{x}(\mathbf{t_i}) = \{\mathbf{x^1}, \mathbf{x^2}, ..., \mathbf{x^m}\}(\mathbf{t_i})$ where $\mathbf{x^k}$ are the relative positions associated to the k_{th} joint with $1 \leq k \leq m$;
- A set of m joint angles: $\mathbf{x}(\mathbf{t_i}) = \{\mathbf{q^1}, \mathbf{q^2}, ..., \mathbf{q^m}\}(\mathbf{t_i})$ where $\mathbf{q^k}$ are the relative rotations associated to the k_{th} joint with $1 \leq k \leq m$;

Every position $\mathbf{x}(\mathbf{t_i})$ is described by $\mathbf{3} \times \mathbf{m}$ dimensional vector for representing the position, or a $\mathbf{4} \times \mathbf{m}$ dimensional vector for representing the quaternionic rotation.

4.2.2 Low-Level Motion Descriptors

Low-level motion descriptors are described as kinematic or dynamic quantities that can be derived directly from the movement representation, or geometric descriptors that describe the skeleton configuration at a given point/points.

Kinematic Descriptors

As seen in the Table 4.1, they are properties of motion like duration of movement, velocity, acceleration, jerk, curvature and quantity of movement.

In the report, bold characters will describe scalar values of quantites (norm).

- **Duration**: describes the length of movement
- **Velocity**: describes velocity for one joint (rate of movement)

$$v^k(t_i) = \frac{x^k(t_{i+1}) - x^k(t_{i-1})}{2\delta t} \tag{4.1}$$

  And its scalar value

$$\mathbf{v^k(t_i)} = \sqrt{v_x^k(t_i)^2 + v_y^k(t_i)^2 + v_z^k(t_i)^2} \tag{4.2}$$

- **Acceleration**: describes velocity for one joint (rate of change of velocity)

$$a^k(t_i) = \frac{x^k(t_{i+1}) - 2x^k(t_i) + x^k(t_{i-1})}{\delta t^2} \tag{4.3}$$

  And its scalar value

$$\mathbf{a^k(t_i)} = \sqrt{a_x^k(t_i)^2 + a_y^k(t_i)^2 + a_z^k(t_i)^2} \tag{4.4}$$

- **Jerk**: describes motion smoothness (rate of change of acceleration)

$$j^k(t_i) = \frac{x^k(t_{i+2}) - 2x^k(t_{i+1}) + 2x^k(t_{i-1}) - x^k(t_{i-2})}{2\delta t^3}$$

(4.5)

  And its scalar value

$$\mathbf{j^k(t_i)} = \sqrt{j_x^k(t_i)^2 + j_y^k(t_i)^2 + j_z^k(t_i)^2} \tag{4.6}$$

- **Curvature**: describes the speed of curve change from a given point. It can be computed as the cross product of acceleration and velocity

$$C^k(t_i) = \frac{||a^k(t_i) \times v^k(t_i)||}{v^k(t_i)^3} \tag{4.7}$$

  And the curvature radius

$$R^k(t_i) = \frac{1}{C^k(t_i)} \tag{4.8}$$

- **Quantity of Motion (QOM)**: describes the weighted average of the velocities of body parts

$$qom(t_i) = \frac{\sum_{k\epsilon K} w_k v^k(t_i)}{\sum_{k\epsilon K} w_k} \tag{4.9}$$

Geometric Descriptors

This descriptors describe geometrical characteristics of the body, relating the body to itself or to the environment. Bounding shapes shown in Figure 4.1 are a good example of geometric descriptors. Geometric descriptors include bounding box, bounding sphere, bounding ellipsoid, convex hull, displacement, rotation and center of mass.

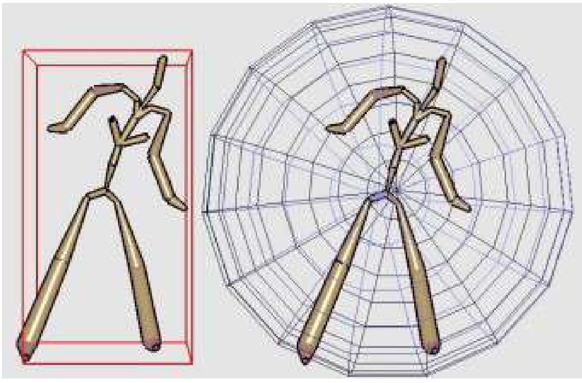


Figure 4.1: *Bounding shapes: Bounding box(left), Bounding sphere(right) [3]*

- **Bounding Box**: describes the box(rectangular parallelepiped) that surrounds the body in 3D (-x, +x, -y, +y, -z, +z)
- **Bounding Sphere**: describes the bounding sphere (center and radius) that surrounds the body in 3D
- **Bounding Ellipsoid**: describes the bounding ellipsoid (center, r_a, r_b, r_c) that surrounds the body in 3D. It is more accurate then the sphere which gives a better volume approximation, but is harder to compute
- **Convex Hull**: describes the convex hull(smallest convex set that contains all joints) that surrounds the body in 3D. It gives a better representation of volume then bounding ellipsoid, but it is harder to compute.
- **Displacement**: describes the distance of the joint or effector k towards the root of the limb, center of mass or the ground. Examples are hand-shoulder, head-root.

$$||x^k(t_i) \times x^l(t_i)|| \tag{4.10}$$

- **Rotation**: describes angular displacement v that shifts the orientation $q_k$of one joint to orientation $q_l$of another joint

$$exp(v^{kl}(t_i)) = q^k(t_i) \cdot q^l(t_i)^{-1} \tag{4.11}$$

- **Center of Mass (COM)**: describes the weighted average of the positions of joints in different body parts

$$com(t_i) = \frac{\sum_{k \epsilon K} w_k x^k(t_i)}{\sum_{k \epsilon K} w_k} \tag{4.12}$$

Effort Descriptors

Effort Descriptors are used to describe the quality of motion related to energy, dynamics and expressiveness. It is consisted from four categories: weight, time, space and flow.

- **Weight Effort**: [8] describes it as a weighted sum of the body's, or body part's, kinetic energy calculated from the joints. Weighted effort is then calculated by estimating the maximum energy of the time interval

$$E(t_i) = \sum_{k \epsilon K} E_k(t_i) = \sum_{k \epsilon K} \alpha_k w_k(t_i) \tag{4.13}$$

- **Time Effort**: describes the sense of urgency This descriptor can by calculated by taking the sum of the accelerations, or the norm of accelerations, of the body or the different body parts over a time period [8]

$$Time^k(t_i) = \frac{1}{T} \sum_{i=1}^{T} a^k(t_i)) \tag{4.14}$$

And for the segment of p joints

$$Time^k(T) = \sum_{k \epsilon K} \alpha_k Time^k(T) \tag{4.15}$$

- **Space Effort**: describes the directness of movement

$$Space^k(T) = \frac{||x^k(t_i) - x^k(t_{i-1})||}{||x^k(T) - x^k(t_1)||} \tag{4.16}$$

And for the segment of p joints

$$Space^k(T) = \sum_{k \epsilon K} \alpha_k Space^k(T) \tag{4.17}$$

- **Flow Effort**: describes the continuity of movement as the aggregated jerk of the body, or body parts during a time period

$$Flow^k(t_i) = \frac{1}{T} \sum_{i=1}^{T} j^k(t_i)) \tag{4.18}$$

And for the segment of p joints

$$Flow^k(T) = \sum_{k \epsilon K} \alpha_k Flow^k(T) \tag{4.19}$$

4.2.3 High-Level Motion Descriptors

In Laban Movement analysis high-level motion descriptors, as seen in Table 4.1, are divided into descriptors relating to the body, the body shape and the space where the body moves. High-level descriptors use low-level features when calculating spatial, kinematic and physical quantities that characterise the movement. This descriptors can be made at various levels of spatial discretization, on the whole body, set of joints or on one joint k.

Body Descriptors

Body descriptors describe the spatial characteristics of the motion and determine the movement of body parts. This component contains the balance, support and the center of mass displacement.

- **Action Presence**: describes if the displacement of the body segment crosses a set threshold

$$Action^k(t_i) = ||x^k(t_i) - x^k(t_{i-1})|| > \epsilon(?1 : 0) \tag{4.20}$$

- **Center of Mass Displacement**: describes displacement of the center of mass in relation to it's starting point

$$com - disp(t_i) = ||com^k(t_i) - com^k(t_{rest})|| \tag{4.21}$$

- **Balance**: describes the Boolean value of the center of mass in relation to the projection of the bounding shape on the floor. If the center of mass is out of the shape, it is considered out of balance
- **Support**: defines the current body segment that is used for support of the body

Space Descriptors

These descriptors relate the movement of the body to the 3D environment which surrounds it. This includes pathways and spatial patterns of the movement. Figure 4.2 refers to the position of the Root joint in the body.

- **Distance Covered**: describes the projected distance of the Root joint on the floor during a certain time period
- **Area Covered**: describes the projected area of the Root joint on the floor during a certain time period
- **Hip Height**: describes the distance between the ground an the Root joint

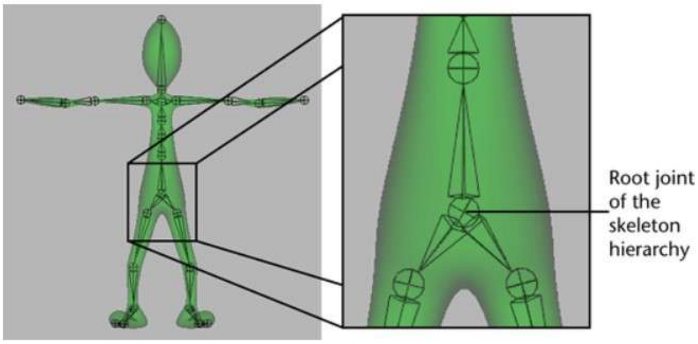


Figure 4.2: *Description of the Root joint in the skeleton*

Shape Descriptors

The shape descriptors are used to describe the changes in the shape of the body during movement. It can be calculated by the bounding shape surrounding the body, or the distances of the end effectors of the body. Shape descriptors consist of shape directional, shape flow and shaping.

- **Bounding Volume**(Shape Flow) describes the volume of the body or body parts, which is computed from the bounding shapes mentioned in the low-level descriptors on Figure 4.1.

- **Shape Directional** describes the path on which the movement is executed. This descriptor associates kinematic and geometric properties of motion. One way of calculating it is defined by calculating the average curvature in 2D, which is obtained trough PCA analysis.

$$ShapeD^k(T) = \sum_{i=1}^{T} C^k(t_i) \tag{4.22}$$

- **Extensiveness**: describes the maximum distance between the body's center of mass and the extremities, like hands, feet, head etc.

$$Extens_m(t_i) = max\ \alpha^k ||com(t_i) - x^k(t_i)|| \tag{4.23}$$

- **Arms Shape**: describes the the distance between the arm system and the body center
- **Elbow Flexion**: describes the elbow flextion angles
- **Shoulder Angle**: describes the angle between the axis defined by the arm and the vertical axis of the torso
- **Hands Relationship**: describes the distance in 3D of the hands
- **Feet Relationship**: describes the distance in 3D of the feet

4.3 Principal Component Analysis

In this report, Principal Component Analysis was used only as a dimensionality reduction tool. As mentioned in [18], PCA is a statistical procedure that uses orthogonal transformation. As an input, it takes in variables that are correlated and gives variables that are uncorrelated. It is usually used to compare the relationship between a group of variables, so it can be used for reducing the dimensions.

Procedure for reduction if assumed that dataset $x^{(1)}, x^{(2)}, x^{(3)}, ...x^{(m)}$ with n dimensions is reduced to k dimensions:

1. **Standardization**: data should have unit variance and zero mean

$$x_j^i = \frac{x_j^i - \overline{x_j}}{\sigma_j} \tag{4.24}$$

2. **Calculating Co-variance Matrix**

$$\sum = \frac{1}{m} \sum_{i}^{m} (x_i)(x_i)^T \tag{4.25}$$

3. **Calculating Eigenvector and Eigenvalue of the Co-variance matrix**

$$u^T \sum = \lambda \mu \tag{4.26}$$

$$U = [u_1 u_2 ... u_n] \tag{4.27}$$

4. **New Data with k Dimensions**

$$x_i^{new} = \begin{bmatrix} u_1^T x^i \\ u_2^T x^i \\ . \\ . \\ . \\ u_k^T x^i \end{bmatrix} \tag{4.28}$$

4.4 K-Nearest Neighbour Algorithm

K-Nearest Neighbours Algorithm (KNN) is a simple, non-parametric, instance-based machine learning algorithm based on Supervised Learning technique. The algorithm works by finding the K number of nearest points to a given test point, and then predicting the class or value of the test point based on the majority class or average value of the K nearest points. The distance between points is typically measured using the Euclidean distance. This means that all of the available data is stored and and the new data points are classified based on similarity. KNN is usually used for Classification problems, but can also be used for Regression. KNN is algorithm that is non-parametric, which means that it does not make any assumptions on the underlying data. It is sometimes referred to as "lazy" because it does not learn from the training set immediately, but instead it stores the data set and preforms the classification at the point of classification.

For example in Figure 4.3, consider the following dataset with two classes, "red" and "green," and two features, "x" and "y." In the plot below, the question mark represents the test point for which we want to predict the class.

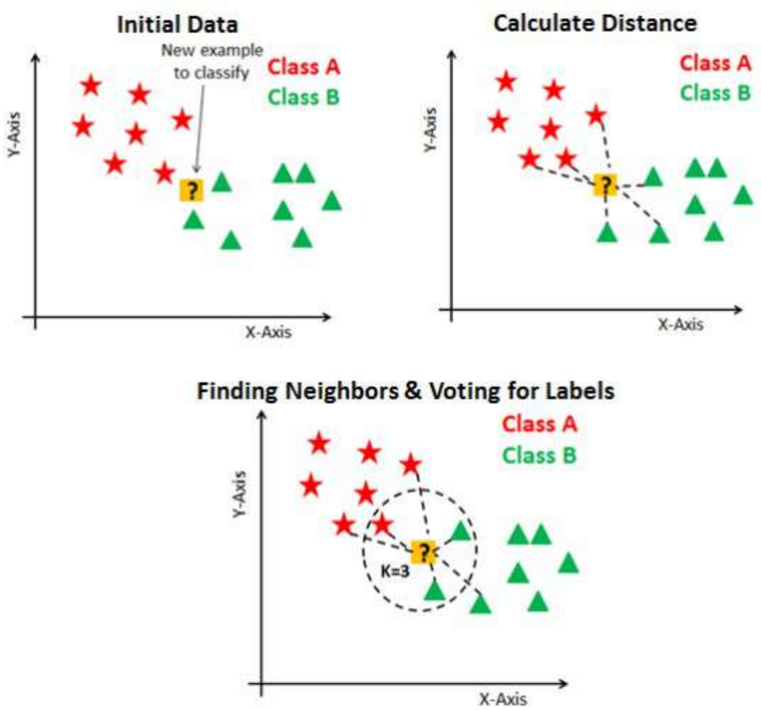


Figure 4.3: *K-NN algorithm example*

If we set K=3, the three nearest points to the test point are two green points and one red point. Since there are more green points among the three nearest neighbors, the algorithm would predict that the test point belongs to the "green" class. On the other hand, if we set K=5, the five nearest points include three red points and two green points. In this case, the algorithm would predict that the test point belongs to the "red" class.

One of the main strengths of KNN is its simplicity and flexibility. It is easy to understand and implement, and it can work well with a small number of features. However, it can also be computationally expensive, especially for large datasets, and it can be sensitive to the choice of K and the distance metric used. [10]

4.5 Support Vector Machines Algorithm

Support vector machines (SVMs) is a supervised learning algorithm that is used for classification and regression tasks. It is based on the principle of locating a hyperplane in a high-dimensional space that maximally separates different classes of data. If the problem was binary, it would separate the classes in two. If the problem requires multi-class classification the algorithm will create multiple binary classifiers and combine them to make predictions. The principle of hyper planes in 2D is shown on Figure 4.4

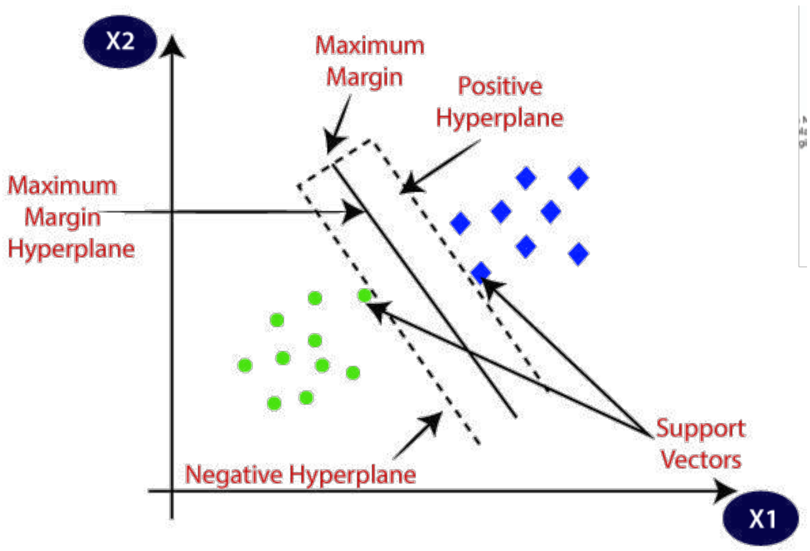


Figure 4.4: *SVM algorithm example*

SVM algorithms are good in their ability to handle high-dimensional data more effectively. So data containing large amount of features is handled by SVM models effectively, since they only use a subset of the training data to build the model. Because of this, if the number of features is larger then the number of samples, the models can still function.

SVM algorithms are robust to overfitting, because of the use of regularization to control the model complexity. The problem is that they can be sensitive to the choice of kernel function and the regularization parameters. Finding the optimal values of these parameters can require experimentation.

There are different types of the SVM models. There are linear SVMs, nonlinear SVMs and kernel SVMs. For data that is linearly separable, linear SVMs are the best, while nonlinear SVMs and kernel SVMs are good for handling more complex data. Choice of the kernel function has a big impapact on the model and it's performance. [11]

5 Data Analysis

In this chapter, the workflow of the project will be presented, together with the data analysis and separation. Because of the fact that this paper is trying to classify general movement, human expression and intent, it was important to separate the data in a way that it can be distinguished from one another. This will help in the classification later. Figure 5.1 shows the workflow of the data analysis.

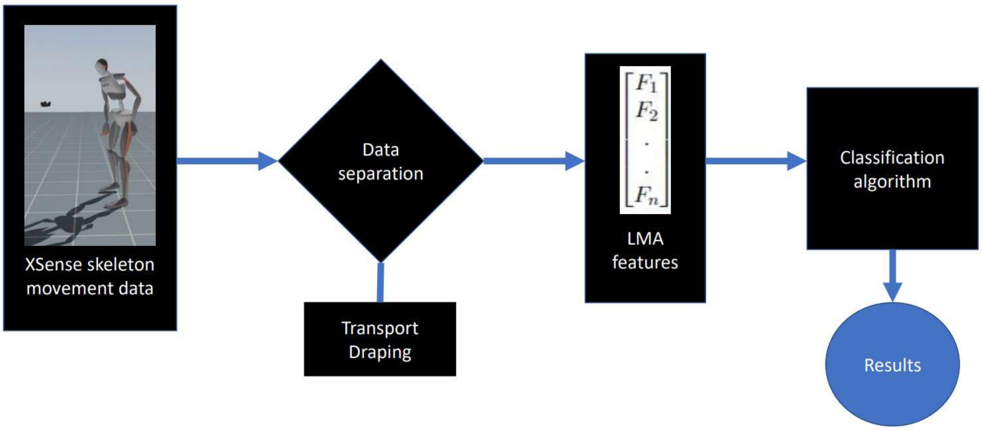


Figure 5.1: *Data analysis workflow*

After the experiment and the data collection, explained in the Chapter 3, a large quantity of unsorted data was gathered. XSense MVN Awinda software was used to convert the data in 40 excel files, which contained relevant and not relevant data. Every participant's data was divided in to the transport and draping task. Data sheet of each participant/experiment contains movement data, as seen from Table 5.1.

Sheet	Types	Data Points
Segment Orientation	Quaternion (q1,q2,q3,q4) Euler (x,y,z)	head, neck, four vertebra (L5, L3, T8, T12), left and right shoulders, upper arms, forearms, hands, upper legs, lower legs feet, toes, pelvis
Segment Position	(x,y,z)	
Segment Velocity	(x,y,z)	
Segment Acceleration	(x,y,z)	
Segment Angular Velocity	(x,y,z)	
Segment Angular Acceleration	(x,y,z)	
Joint Angles	(x,y,z)	joints of the body
Joint Ergonomic Angles	(x,y,z)	
Center of Mass	(x,y,z)	CoM position CoM velocity CoM acceleration

Table 5.1: *Excel sheet of the collected movement data*

The first division of data was made in terms of the task. Data was separated on data related to transport task, and data related to draping task. Visualisation of the data separation can be seen in Figure 5.2.

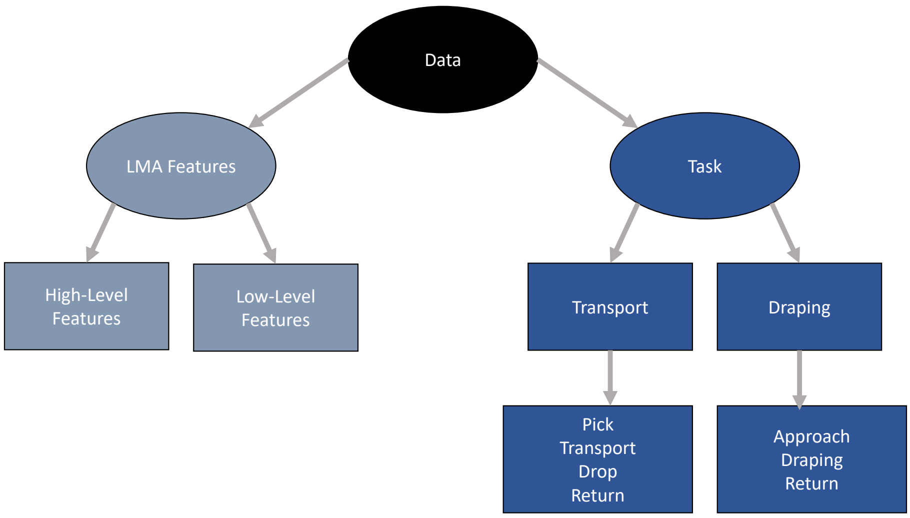


Figure 5.2: *Data separation*

As can be seen, the data was separated in two steps. One step was to calculate the Laban Movement Analysis features by following the method from Section 4.2. As explained in the earlier chapter, this divides the features into high-level and low-level descriptors. The other division step was to separate the data by the task and by the specific movement that is happening inside of the task.

5.1 Data Separation by Task

As mentioned, and seen in Figure 5.2, the data separation based on task is divided to the transport and draping. After looking for a way to separate the data in a standardised way related to the movement inside of the task, the conclusion was to look at the movement data as a signal. One of the XSense MVN Awinda sensors was placed on the end effector of the robotic arm that was doing the collaborative task with the participants. This sensor captured the information in a same way as the body sensors. Figure 5.3 shows this sensor.

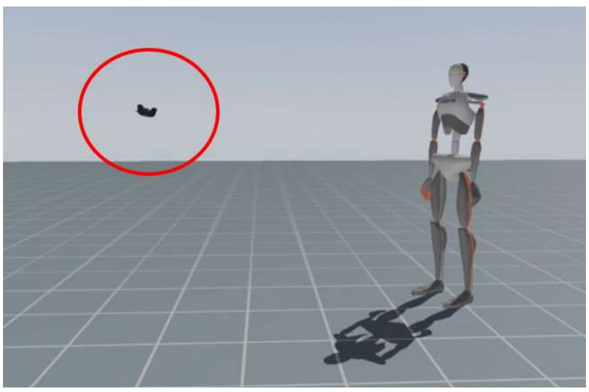


Figure 5.3: *Sensor of the robotic arm*

Movement of this sensor in the X-axis was used as the indicator of the task. As shown in Figure 5.4, peaks and minimums of the robot movement in X-axis was found with the help of the second derivation of the movement positions in the X-axis. After the needed minimums and peaks were found, the whole movement was divided. Figure 5.4 shows the division of draping on draping, approach and return.

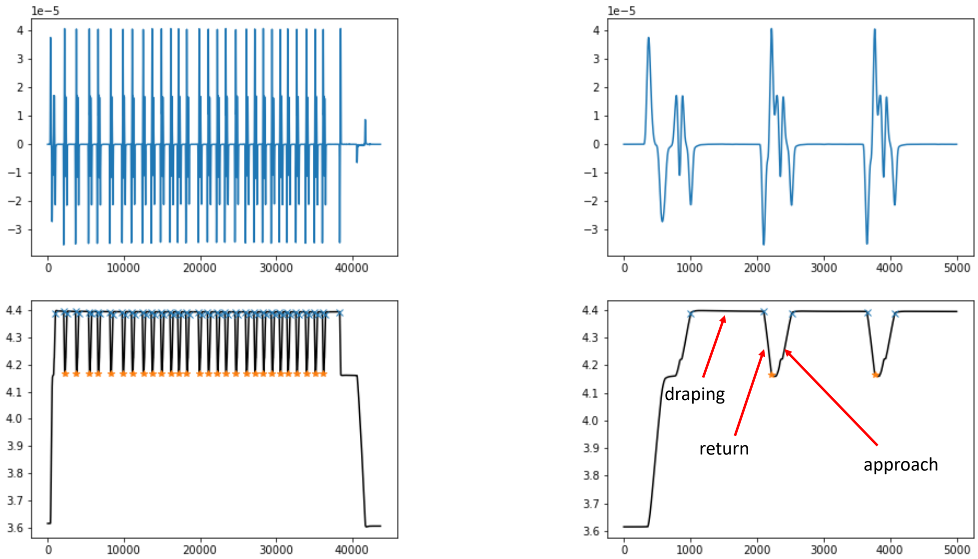


Figure 5.4: *Robot movement separation in X-axis (draping)*

Transport was divided in to pick, transport, drop and return by the same principle. This is shown in the following Figure 5.5.

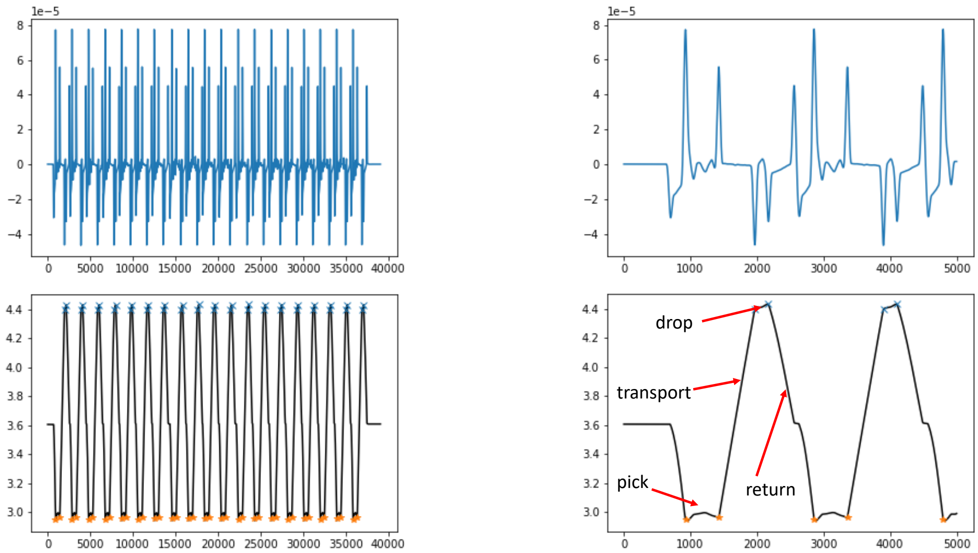


Figure 5.5: *Robot movement separation in X-axis (transport)*

5.2 Feature Extraction

This section will show the features that will be used in the classification machine learning algorithms. All of the features are calculated based on the Laban Movement Analysis, which were explained in the Section 4.2. The approach trough the feature extraction process was to fit the equations of the descriptors to the data gathered from the movement, or more specifically from the sensors placed on the body of each participant. Like the Table 4.1, a new table of calculated descriptors/features was made, and shown in Table 5.2.

Low-Level	**High-Level**
Kinematic / Dynamic Descriptors	Body Descriptors
Duration *Velocity* *Acceleration* *Jerk* *Curvature* *Quantity of Motion*	*Center of Mass-Displacement* *Balance*
Geometric Descriptors	Space Descriptors
Bounding box *Bounding ellipsoid* *Displacement* *Center of Mass*	*Distance covered* *Hip height*
Effort Descriptors	Shape Descriptors
Time effort *Flow effort*	*Bounding volume* *Extensievness* *Shape directional*

Table 5.2: *Table of calculated features*

Limitations

Before showing the calculated features it is important to note that the Laban Movement Analysis paper, shown in [3], in which the descriptors were described, explained all of the descriptor calculations in terms of joints and joint rotations as can be seen in Section 3.4. In case of this experiment, the same data was not available to have this kind of approach. XSense MVN Awinda motion tracking suit does not have locations of the joints or joint velocities and accelerations in the 3D environment. However, for most of the descriptors the rotation is not important, so instead of the joint locations, velocities and accelerations descriptors were made with segments. Also, since the XSense trackers can measure velocity and acceleration, those measures were used instead of the location parameters.

5.2.1 Low-Level Features

In this Section, the low-level features are presented. As seen from the Table 5.2, 12 descriptors were calculated, which resulted in 187 features.

Kinematic Descriptors

- **Duration**: this descriptor was not calculated, because the motion tracking devices record the motion by 60 Hz in frames. That describes the duration of the movement.
- **Velocity**: velocity was recorded by the motion tracking software and the norm of velocity was calculated, as shown in Figure 5.6

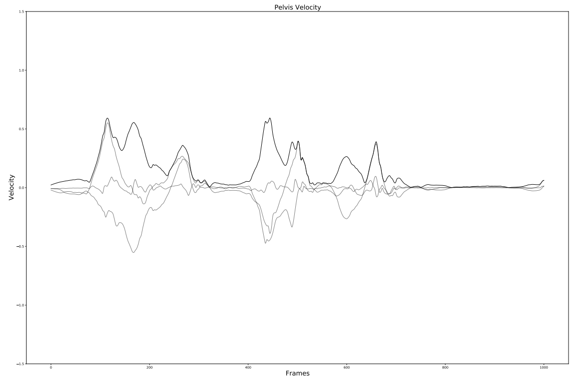


Figure 5.6: *Pelvis velocity norm (bold), pelvis velocity (x,y,z)*

- **Acceleration**: acceleration was recorded by the motion tracking software and the norm of acceleration was calculated, as shown in Figure 5.7.

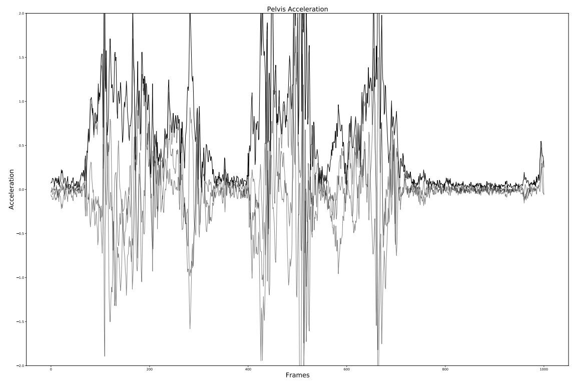


Figure 5.7: *Pelvis acceleration norm (bold), pelvis acceleration (x,y,z)*

- **Jerk** Jerk movement, which is a third derivation of position was calculated by the acceleration and frames. This correlates to the Formula 4.2.2, but uses the information about the acceleration instead of position. It was calculated for all of the 23 body segments.

$$j(t_i) = \frac{a_{i+1} - a_{i-1}}{t_{i+1} - t_{i-1}} \tag{5.1}$$

Norm of velocity, acceleration and jerk were calculated for all of the 23 body segments.

- **Curvature**: calculations were done by using the Formula **??** with the sensor velocity and acceleration data. It was calculated for all of the 23 body parts.
- **Quantity of Motion**: this descriptor was calculated for the whole body with weights of different body parts set on 1. This resulted in 1 feature for the whole body.

Geometric Descriptors

- **Bounding Box and Bounding Ellipsoid**: to calculate these features, positions in 3D were used. First, the center of the box (rectangular parallelepiped) and ellipsoid was defined by the location of the pelvis. Then the segments of the body with the maximum and minimum distance from the sensor (center of the coordinate system) in x,y and z direction were found. The subtraction of the pelvis distance from the sensor in x,y and z direction towards the maximum and minimum points of the body was defined as the a, b and c sides of the box or the r_a, r_b, r_c radii of the ellipsoid. This measures describe only the distances of the radii or box sides, not the volume. This resulted in 6 features.

- **Displacement**: determining the displacement was done by calculating the euclidean distance between the different segments of the body. For the purpose of this research, displacement was determined for: Left Hand - Left Shoulder, Right Hand - Right Shoulder, Right Hand - Left Hand, Head - Body Center of Mass, Right Foot - Body Center of Mass, Left Foot - Body Center of Mass and for Right Hand - Left Hand. This resulted in 7 features.

$$D = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2} \tag{5.2}$$

- **Center of Mass**: position of the center of mass was already provided by the motion tracking software, so that resulted in 1 feature.

Effort Descriptors

- **Time Effort and Flow Effort**: the features were calculated for the whole body (all segments), left and right arm (hand segment, forearm segment, upper arm segment and shoulder) and feet (upper leg, lower leg, feet and toes). This resulted in 8 features.

5.2.2 High-Level Features

In this Section, the high-level features are presented. As seen from the Table 5.2, 6 descriptors were calculated, which resulted in 31 features.

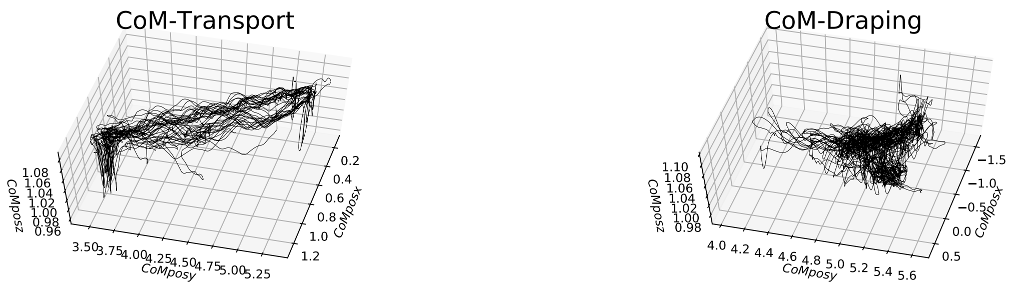
Body Descriptors

- **Center of Mass - Displacement**: $com(t_{rest})$ was defined as the first recorded frame of the center of mass position. Following the same principle as the displacement formula, shown in Formula 5.2, euclidean distance of the center of mass was calculated in respect to it's rest position. This resulted in 1 feature.

- **Balance**: balance was calculated with the ellipsoid projection on the floor. If the center of mass of the body was out of the projection balance was 0, and if it was inside balance was 1. This created 1 feature, in boolean form.

Space Descriptors

- **Distance Covered**: distance covered was calculated as the cumulative sum of distances between the frames. This resulted in 1 feature.



Distance covered during transport *Distance covered during draping*

Figure 5.8: *Distance covered in 3D*

- **Hip Height**: the height of the pelvis in z-axis provided by the movement sensors was taken as hip height

Shape Descriptors

- **Bounding Volume**: the volume of the box and the ellipsoid was calculated with the measures from the low-level geometric descriptors. The box sides and the radii were used to calculate the volume by the volume formulas. The comparison between volumes is shown in Figure 5.9. This resulted in 2 features.

$$V_{ellipsoid} = \frac{4}{3}\pi r_a r_b r_c \tag{5.3}$$

$$V_{box} = abc \tag{5.4}$$

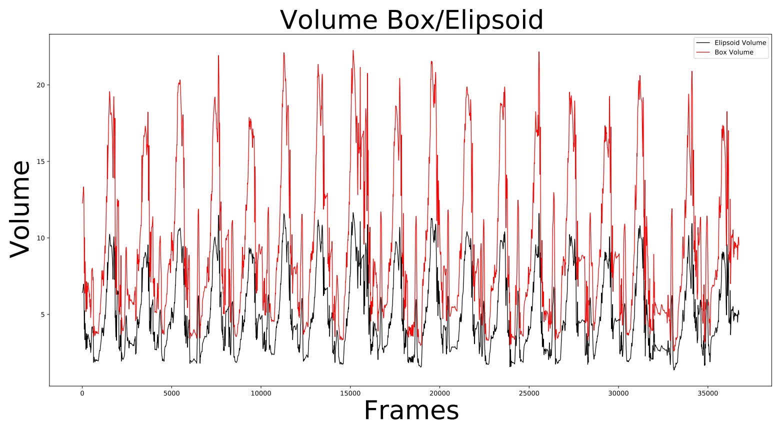


Figure 5.9: *Box (red) and ellipsoid (black) volume graph*

- **Extensiveness**: this descriptors was calculated with the values gathered from the geometric descriptors and subtracted from the position of the center of mass. That way the maximum distance between extremities and the center of mass is calculated. This was calculated for the whole body which resulted in 1 feature.

- **Shape Directional**: as explained in 4.2, one way of calculating the shape directional is to use PCA analysis, explained in 4.3, to go from 3D (x,y,z) to a 2D space while retaining as much variance as possible. After that two new variables were used to calculate the shape directional with the Formula 4.22. This was done for all of the body segments and resulted in 23 new features.

5.3 Trust scores

The trust results from the questioners will be used as labels in the machine learning algorithms. Every participant did 3, 14 question questioners and rated the trust in the robot. The trust was evaluated before the experiments, after the transport task and after the draping task. Table 5.3 shows the trust scores of the first 3 participant while excluding the trust evaluation before the experiments, because it did not look like a good representation of the trust. The median of the trust scores for all participants was calculated, and the trust was binary classified as low (under the median) or high (over the median), which will be used as the label for classification.

Subject	**TrustScore**	**Task**
1	0.5	Draping
1	0.45	Transport
2	0.7	Transport
2	0.707143	Draping
3	0.535714	Draping
3	0.492857	Transport

Table 5.3: *Trust scores*

6 Results

Before testing the models, high-level and low-level features of 20 participants were merged together in to dataframes for specific movements in the transport task and the draping task. This resulted in 7 dataframes for the high-level features and 7 dataframes for the low-level features.

After the data separation and analysis, different supervised machine learning models were used for classification of the gathered data. After the testing of the algorithm, its performance will be evaluated and stated in the report in the form of a confusion matrix, precision, recall and F1 score.

Confusion matrix is a table that is used to evaluate the performance of a classification model and it is a summary of the model's predictions compared to the true values. Confusion matrix contains four measures:

- true positive (TP): number of instances where the positive class is correctly predicted as positive
- false positive (FP): number of instances where the positive class was incorrectly predicted as positive.
- false negative (FN): number of instances where the negative class is incorrectly predicted
- negative (TN): number of instances where the negative class is correctly predicted

These measures are used to calculate the metrics needed to express the performance of the model:

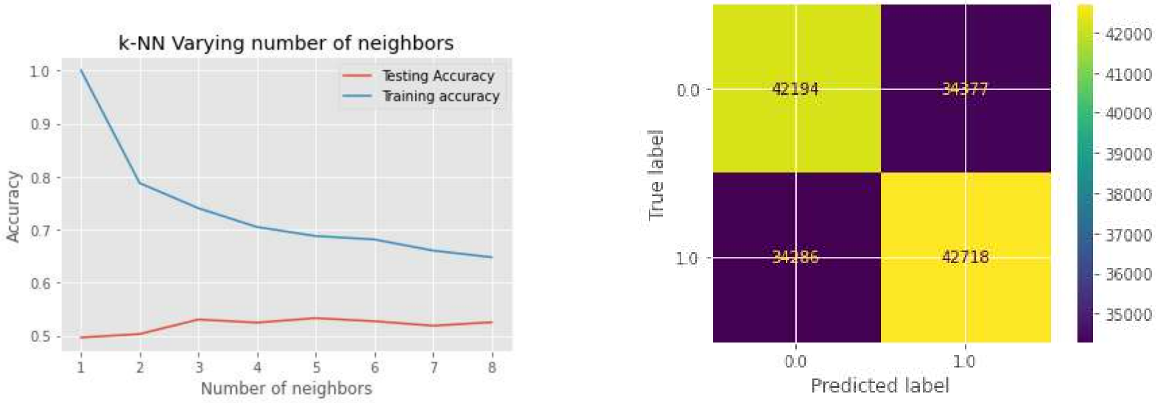
- **Recall**: this is a measure of the model's ability to correctly identify all instances of the positive class $Recall = \frac{TP}{TP+FN}$
- **Precision**: this is a measure of the model's ability to correctly classify the positive class

  $Precision = \frac{TP}{TP+FP}$
- **F1 Score**: metric that combines precision and recall. It is defined as the harmonic mean of precision and recall

  $F1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}$

6.1 K-Nearest Neighbours - Results

As a start, K-Nearest Neighbours was used. First test used number of neighbours n = 3. It resulted in the algorithm accuracy close to 55%. Figure 6.1 shows the confusion matrix of this KNN algorithm. Since the accuracy of 55% was not good, idea was to test different parameters of the algorithm. Because of that a relation between the number of neighbours and algorithm precision was made. Figure 6.1 shows that relation between the training accuracy and the testing accuracy. Because of this, another instance of the algorithm was run, with a higher number of neighbours to compare the results. Table 6.1, shows a comparison between the accuracy scores of 3 draping processes and 1 transport process and between the numbers of neighbours. This data was made by the input of only low-level features.



N-neighbours influence *Confusion matrix*

Figure 6.1: *Number of neighbours (left) and confusion matrix (right)*

n -neighbours	**3**	**15**
Draping -process of draping	*accuracy score* = 55.29% *precision score* = 55.4% *recall score* = 55.48% *F1 score* = 55.44%	*accuracy score* = 55.9% *precision score* = 55.9% *recall score* = 57.13% *F1 score* = 56.5%
Draping -process of approach	*accuracy score* = 55.42% *precision score* = 58.26% *recall score* = 62.1% *F1 score* = 60.3%	*X*
Draping -process of return	*accuracy score* = 55.8% *precision score* = 59.9% *recall score* = 60.83% *F1 score* = 60.36%	*X*
Transport -process of transport	*accuracy score* = 56.8% *precision score* = 55.9% *recall score* = 57.13% *F1 score* = 56.5%	*X*

Table 6.1: *Comparison of KNN accuracy*

This comparison showed that between the different tasks of transport and draping, there was no significant influence in the algorithm accuracy. Also the number of neighbours did not affect the accuracy.

PCA dimensionality reduction was used to get a visual of the 31 features of high-level data in two dimensions, which is shown in the Figure 6.2.

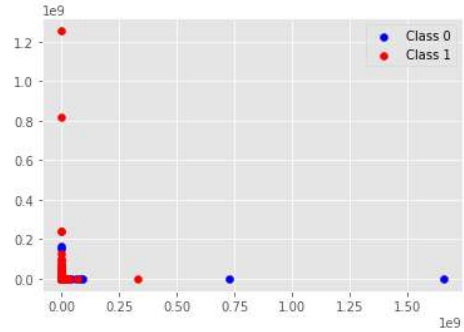


Figure 6.2: *PCA dimensionality reduction*

Having such a large amount of features together close to 0, can be an indicator that there is a lot of noise in the data, and that a lot of information can be redundant. Also, after implementing batch approach with an overlap to the KNN algorithm, which resulted in lower accuracy scores for the high-level and low-level features, the number of features looked like a problem. Since the HRI team with previous experiments had an idea to look at which features could be more important to look at, the next step was to look at datasets with less features.

From the low-level features, all of the features except from the displacement ones were removed. Only features that were left were the 6 displacement features:

- Left Hand - Left Shoulder
- Right Hand - Right Shoulder
- Right Hand - Left Hand
- Head - Body Center of Mass
- Left Foot - Body Center of Mass
- Right Foot - Body Center of Mass

From the high-level features, shape directional descriptor, which had 23 features was removed, while keeping all the other descriptors. This was done because the shape directional descriptor had a lot of data variety and noise. That left the high-level features with 5 features:

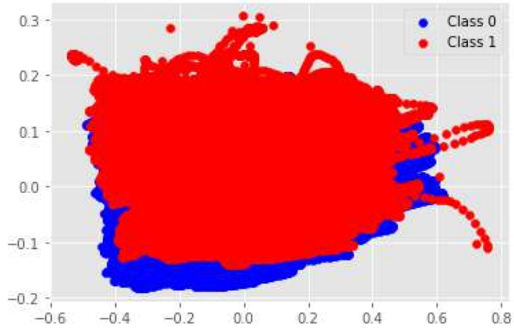
- Center of Mass - Displacement
- Bounding Volume Box
- Bounding Volume Ellipsoid
- Pelvis Distance Covered
- Head Distance Covered
- Extensiveness

Implementing new reduced amount of low and high-level features in the KNN algorithm yielded in much better results. Figure 6.3 shows the confusion matrix and the PCA analysis of reduced low-level descriptors.

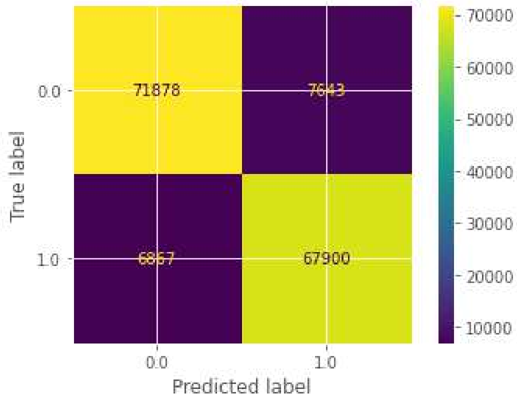
KNN algorithm accuracy, with reduced low-level descriptors for the draping task:

- accuracy score = 90.6%
- precision score = 89.8%

- recall score = 90.8%
- F1 score = 90.34%



N-neighbours influence



Confusion matrix

Figure 6.3: *PCA analysis (left) and confusion matrix (right)*

Removing the number of features in the dataset showed significant improvement in the accuracy of the algorithms and looks like a good way forward in the analysis.

It is important to note that the KNN algorithm showed results that had a high difference in accuracy. The algorithm with reduced amounts of features seemed too idealistic, so the next step was to try different algorithms.

6.2 Support Vector Machines - Results

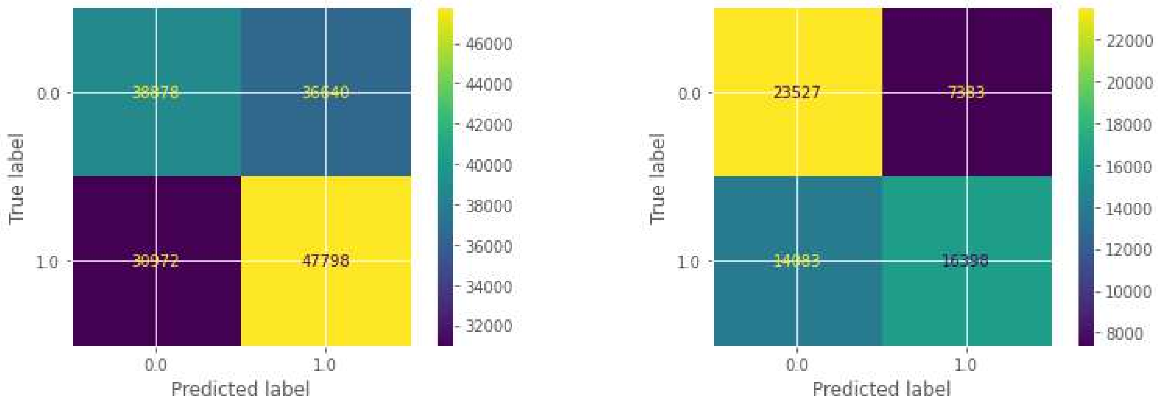
The Support Vector Machines algorithm, as explained in Section 4.5, is good in handling larger datasets witch have a larger number of features. Because of the too high compiling times of nonlinear SVM algorithms, a linear approach was selected, with a batch approach that includes overlapping. Since KNN showed significant improvements in handling data, where the number of features was reduced, same datasets with reduced features were selected for analysis.

A comparison between the low-level and high-level reduced features of the same draping task was made with the SVM model. Table 6.2 shows the comparison in terms of accuracy of the models.

Model Accuracy	**SVM (Low-Level Features)**	**SVM (High-Level Features)**
accuracy score	*56.18%*	*65%*
precision score	*56.6%*	*68.95%*
recall score	*60.68%*	*60.44%*
score	*58.57%*	*53.79%*

Table 6.2: *Comparison of SVM models for low and high-level descriptors*

This comparison could be an indicator that the low-level features of displacement are worse then some of the high-level features at correlating the trust to the movement. Figure 6.4 shows the comparison between the confusion matrices of the two SVM models.



Low-Level *High-Level*

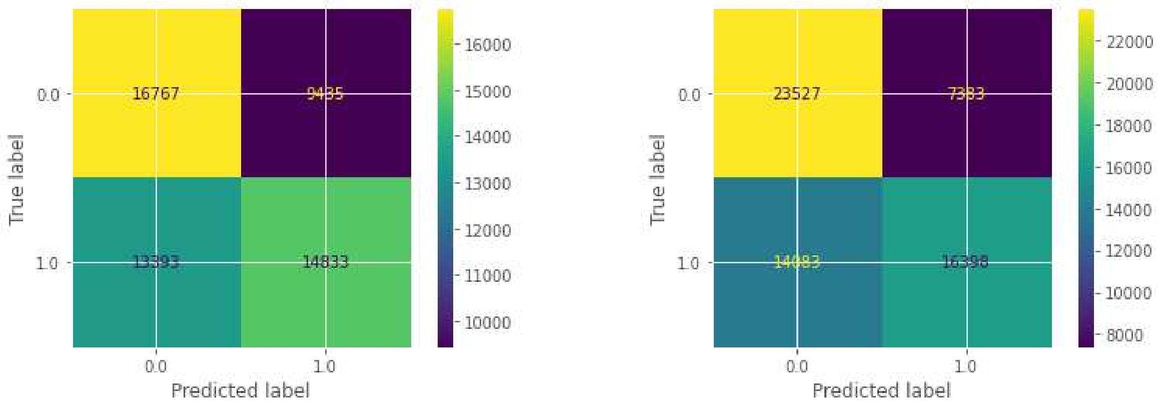
Figure 6.4: *Confusion matrices for low and high-level descriptors*

Afterwards, a comparison has been made in terms of the different task. High-level features of draping and transport were compared and Table 6.3 shows the results.

Model Accuracy	**SVM Transport**	**SVM Draping**
accuracy score	*60.06%*	*65%*
precision score	*61.12%*	*68.95%*
recall score	*55.5%*	*60.44%*
score	*56.51%*	*53.79%*

Table 6.3: *Comparison of SVM models for transport and draping*

Results indicate that there is not a big difference between different tasks influencing the accuracy precision. Draping task has better accuracy by a small amount. Figure 6.5 shows the confusion matrices of the comparison.



Transport *Draping*

Figure 6.5: *Confusion matrices for transport and draping*

7 Discussion

Movement-Trust Correlation

The conclusion of analysis showed a low correlation of movement descriptors with the measured trust of the participants when implementing a large number of descriptors on the KNN algorithm. After the feature reduction, the accuracy of correlation was improved with the KNN algorithm, but SVM showed similar accuracy results in both cases, which showed a small potential, which would require further research and experiments.

Descriptors

As mentioned in [3], Laban Movement Analysis is good when describing specific movement patterns which happen in dance and choreography, but a lot of descriptors were not usable in this research. Some of the examples were Balance and Support, which did not make sense in repetitive movements like in the experiments. Participants were never out of balance, and only support segments that were used were the feet on the floor. In reality, it is hard to capture expressions and emotions by using motion tracking. Implementing more sensors that would measure features like the hearth rate could prove beneficial.

SVM analysis also showed an indication that the high-level features could be better at evaluating the trust correlation then the low-level ones.

Trust Label

The first obvious thing to look at was the labels used as trust indicators. In machine learning the grand truth (also known as correct answers or true labels) do not need to be perfectly precise. Often the grand truth can be noisy or uncertain because of different factors, which include: error, subjectivity or variability in the data. In this case subjectivity could be a big problem, together with the low amount of data on trust.

In this case motion tracking software recorded big datasets, but participants were asked to measure their trust in the robots only 3 times per session. The variability in the results of that trust is also very low, which could lead to wrong correlation results. If the ground truth is incorrect or unreliable, it can lead to poor model performance and inaccurate predictions. Therefore, it is important to carefully consider the quality and reliability of the ground truth when using it to train and evaluate machine learning models. A solution to that would be to collect additional data in future tests, or devise a better way to collect that data.

One more thing to mention is that the trust relationship evolves during time. In this experiment trust was measured only after each task, so a lot of data was lost inside of the tasks.

Number of Features

In terms of the accuracy increasing after using a smaller number of features. The relationship between the number of features and accuracy of the model can be complex and rely on different factors like the quality of features. This analysis showed that low-level features, especially the Displacement descriptor, have a significant impact on correlation compared to other ones. Also it showed that high-level descriptor Shape Directional influences accuracy in a bad way, because of the bad quality of the data. Using a bigger amount of features increases the complexity of the model, because more information is available. This can lead to overfitting of the model, which means it will be too sensitive to training data, and perform badly on new data. If a smaller amount of features is used, this will simplify the model and make it less prone to overfitting, but because of less information it may be less powerfull if the relevant features were removed. The key is to select the features with the best influence on the overall performance of the model. It is a good idea to try different features and feature combinations and see what works best.

Using Different Algorithms

For this project two classification algorithms were selected. When selecting different algorithms, there are important things to consider.

One of those things is the nature of the data. Some models are more effective when they handle data that is high-dimensional, while some are better for data with a large number of features. It is important to understand what type of algorithm is suitable for the specific problem.

On the other hand, the computation time and resources available can influence the decision between different models. Some algorithms will require more time to compile the data then others. In this case, the SVM algorithm, even thou a linear one was used, required much more computation time then the KNN. But it gave more consistent accuracy results, when implementing more and less features.

Most important thing to consider in the end is the performance of the model. It is important to evaluate performance on different algorithms to choose the one that performes the best.

Bibliography

[1] J.-L. B. Abdelhakim Cherouat Houman Borouchaki. "Geometrical and mechanical draping of composite fabric". In: *Revue Européenne des Éléments Finis* (2005).

[2] S. A. Barbic J. "Segmenting motion capture data into distinct behaviors." In: *Proceedings of Graphics Interface* (2004).

[3] S. G. Caroline Larboulette. "A Review of Computable Expressive Descriptors of Human Motion". In: (2016).

[4] E. A. Catherine Pelachaud Jean-Claude Martin. *Intelligent Virtual Agents*. Springer, 2007.

[5] J. B. L. Chang S.Nam. *Trust in Human-Robot Interaction*. Academic Press, 2021.

[6] R. P. L. Chiuhsiang Joe Lin. "Classification of mental workload in Human-robot collaboration using machine learning based on physiological feedback". In: *Journal of Manufacturing Systems* (2022).

[7] J. O. Fod A. Mataric M.J. "Automated derivation of primitives for movement classification." In: *Autonomous Robots* (2002).

[8] Y. M. Hachimura K. Takashina K. "Analysis and evaluation of dancing movement based on lma." In: *IEEE Intl. Workshop on Robot and Human Interactive Communication* (2005).

[9] HORIZON2020. *Human-robot collaboration on draping of composite parts*. https://cordis.europa.eu/project/id/101006732. Accessed on 2022-11-12. Dec. 2022.

[10] JavaTPoint. *K-Nearest Neighbor(KNN) Algorithm for Machine Learning*. https://www.javatpoint.com/k-nearest-neighbor-algorithm-for-machine-learning. Accessed on 2022-14-12. Dec. 2022.

[11] JavaTPoint. *Support Vector Machine Algorithm*. https://www.javatpoint.com/machine-learning-support-vector-machine-algorithm. Accessed on 2022-14-12. Dec. 2022.

[12] JavaTpoint. *Supervised Machine Learning*. https://www.javatpoint.com/supervised-machine-learning#:~:text=Supervised%20learning%20is%20the%20types,tagged%20with%20the%20correct%20output.. Accessed on 2022-11-12. Dec. 2022.

[13] B. S. Kadir Alpaslan Demira Gözde Dövena. "Industry 5.0 and Human-Robot Coworking". In: *Mugla Journal of Science and Technology* (2017).

[14] P. S. Kahol K. Tripathi P. "Automated gesture segmentation from dance sequences." In: *Automatic Face and Gesture Recognition.* (2004).

[15] . B. G. Lee N. T. Resnick P. "Algorithmic bias detection and mitigation: Best practicesand policies to reduce consumer harms." In: *Brookings Institute: Washington, DC, USA.* (2019).

[16] E. Lee. "How do we build trust in machine learning models?" In: *SSRN Electronic Journal* (2021).

[17] D. R. B. Peter A. Hancock. "A Meta-Analysis of Factors Affecting Trust in Human-Robot Interaction". In: *Human Factors: The Journal of the Human Factors and Ergonomics Society* (2011).

[18] G. T. Reddy et al. "Analysis of Dimensionality Reduction Techniques on Big Data". In: *IEEE Access* (2020).

[19] K. E. Schaefer. "Measuring Trust in Human Robot Interactions: Development of the "Trust Perception Scale-HRI"". In: *Robust Intelligence and Trust in Autonomous Systems* (2016).

[20] T. B. Sheridan. "Human–Robot Interaction: Status and Challenges". In: *SAGE Journals* (2016).

[21] M. T.Arai R.Kato. "Assessment of operator stress induced by robot collaboration in assembly". In: *CIRP Annals* (2010).

[22] M. S. Theresa Law. "Trust: Recent Concepts and Evaluations in Human-Robot Interaction". In: (2020).

[23] Vectornav. *What is an internal measurement unit?* https://www.vectornav.com/resources/inertial-navigation-articles/what-is-an-inertial-measurement-unit-imu. Accessed on 2022-14-12. Dec. 2022.

[24] Wikipedia. *HRI*. https://en.wikipedia.org/wiki/Human%E2%80%93robot_interaction. Accessed on 2022-29-11. Nov. 2022.

[25] Wikipedia. *Isaac Asimov*. https://en.wikipedia.org/wiki/Isaac_Asimov. Accessed on 2022-25-11. Nov. 2022.

[26] T. Yoshikawa. "Machine learning for human movement understanding". In: *Advanced robotics* (2020).

[27] B. N. Zhao L. "Acquiring and validating motion qualities from live limb gestures." In: *Journal of Graphical Models* (2005).

A Trust Questioner

Interpretation of the 40 item scale. The 40 item human-robot trust scale provided an overall percentage score across all items. Items were preceded by the question "What percentage of the time will this robot …" followed by a list of the items. Each item was a single word or short phrase, and the order of items was randomized for each participant. The finalized 40 item scale is provided in Table 35, and took between 5-10 minutes to complete.

Table 35

Finalized Trust Scale

What % of the time will this robot…	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
Act consistently*	o	o	o	o	o	o	o	o	o	o	o
Protect people	o	o	o	o	o	o	o	o	o	o	o
Act as part of the team	o	o	o	o	o	o	o	o	o	o	o
Function successfully*	o	o	o	o	o	o	o	o	o	o	o
Malfunction ^{R}*	o	o	o	o	o	o	o	o	o	o	o
Clearly communicate	o	o	o	o	o	o	o	o	o	o	o
Require frequent maintenance ^{R}	o	o	o	o	o	o	o	o	o	o	o
Openly communicate	o	o	o	o	o	o	o	o	o	o	o
Have errors ^{R} *	o	o	o	o	o	o	o	o	o	o	o
Perform a task better than a novice human user	o	o	o	o	o	o	o	o	o	o	o
Know the difference between friend and foe	o	o	o	o	o	o	o	o	o	o	o
Provide Feedback*	o	o	o	o	o	o	o	o	o	o	o
Possess adequate decision-making capability	o	o	o	o	o	o	o	o	o	o	o
Warn people of potential risks in the environment	o	o	o	o	o	o	o	o	o	o	o
Meet the needs of the mission*	o	o	o	o	o	o	o	o	o	o	o
Provide appropriate information*	o	o	o	o	o	o	o	o	o	o	o
Communicate with people*	o	o	o	o	o	o	o	o	o	o	o
Work best with a team	o	o	o	o	o	o	o	o	o	o	o

139

Figure A.1: *Trust questioner [19]*

What % of the time will this robot…	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
Keep classified information secure	o	o	o	o	o	o	o	o	o	o	o
Perform exactly as instructed*	o	o	o	o	o	o	o	o	o	o	o
Make sensible decisions	o	o	o	o	o	o	o	o	o	o	o
Work in close proximity with people	o	o	o	o	o	o	o	o	o	o	o
Tell the truth	o	o	o	o	o	o	o	o	o	o	o
Perform many functions at one time	o	o	o	o	o	o	o	o	o	o	o
Follow directions*	o	o	o	o	o	o	o	o	o	o	o

What % of the time will this robot be …	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
Considered part of the team	o	o	o	o	o	o	o	o	o	o	o
Responsible	o	o	o	o	o	o	o	o	o	o	o
Supportive	o	o	o	o	o	o	o	o	o	o	o
Incompetent ^{R}	o	o	o	o	o	o	o	o	o	o	o
Dependable *	o	o	o	o	o	o	o	o	o	o	o
Friendly	o	o	o	o	o	o	o	o	o	o	o
Reliable *	o	o	o	o	o	o	o	o	o	o	o
Pleasant	o	o	o	o	o	o	o	o	o	o	o
Unresponsive ^{R} *	o	o	o	o	o	o	o	o	o	o	o
Autonomous	o	o	o	o	o	o	o	o	o	o	o
Predictable *	o	o	o	o	o	o	o	o	o	o	o
Conscious	o	o	o	o	o	o	o	o	o	o	o
Lifelike	o	o	o	o	o	o	o	o	o	o	o
A good teammate	o	o	o	o	o	o	o	o	o	o	o
Led astray by unexpected changes in the environment	o	o	o	o	o	o	o	o	o	o	o

Note. The 14 trust subscale items are marked with an *.
The R represents reverse coded items for scoring.

140

Figure A.2: *Trust questioner [19]*