UNIVERSITY OF BIELEFELD

BACHELOR THESIS

Efficient Target Identification during Haptic Search in a Three-Dimensional Environment

Author:
Julian Nowainski

Supervisor: Dr. Alexandra Moringen
Second Supervisor: Dr. Malte Schilling

A thesis submitted in fulfillment of the requirements for the degree of Bachelor of Science

in the

Neuroinformatics Group
Cluster of Excellence Cognitive Interaction Technology (CITEC)

October 29, 2017

Declaration of Authorship

I, Julian Nowainski, declare that this thesis titled, "Efficient Target Identification during Haptic Search in a Three-Dimensional Environment" and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed: _____

Date: _____

University of Bielefeld

Abstract

Faculty of Technology
Cluster of Excellence Cognitive Interaction Technology (CITEC)

Bachelor of Science

Efficient Target Identification during Haptic Search in a Three-Dimensional Environment

by Julian Nowainski

In this work the goal was to analyze the human way of efficient target identification during a haptic search in a three-dimensional environment. Therefore an experiment was proposed in which blindfolded participants were asked to localize a target object on a modular haptic stimulus board (MHSB) among different items. Both the target and distractor objects were wooden bricks of five different tactile shapes whereof multiple objects of each shape were embedded in a configurable wooden board. The participants had to perform this task in different scenarios each with its own distribution of objects and a different target to search for. During this experiment multimodal data was recorded with a glove capable of capturing both haptic data and joint angles between the fingers.
By performing multiple classification tasks it should be investigated how the data of same stimuli correlates to their role as a target or distractor object in the scenarios. The hypothesis was that the same object would yield different data in the cases where it was searched for and where it was a distractor.
It was found that from the data models could be build to distinguish between these roles and it was also possible to show that these models performed better when they were trained only on the data from targets. This means that the efficiency of searching a target is based on a set of salient features that is sufficient enough to differentiate between the target or a distractor role, but not necessarily between the objects itself.

Contents

Chapter 1

Introduction

1.1 Motivation

Humans are very skilled when it comes to the task of exploring objects based merely on the haptic feedback they get from touching them. In a setting with multiple objects, a desired object can be found quite fast among the others. Searching for some objects like keys in your bag, or a phone on your nightstand in the dark are some examples of a three-dimensional haptic search task. In such tasks, one searches for an desired object called the target among other objects that are not of interest called distractors. Often, it is a specific feature of an object that makes it stand out among others. These features can be, for an instance, material properties, size, wight or shape [6]. This is called the pop-out effect where the target feature is then said to be salient with respect to the distractor properties.
This phenomenon has not only been aspect of research in the haptic domain, but rather had its beginning in the vision. An example was to find a red dot among green ones, which could be done effortlessly without the need for a throughout search. However, finding a line with a specific rotation in an image with multiple lines of various rotations takes some effort [11]. Furthermore Ledermann and Klatsky found in 1987 [8], that each search strategy consists of a set of patterns of explorations that are called exploratory procedures (EPs). This means that the efficiency of a haptic search is not only based on the salient target feature, but also on the search strategy that is used. This shows the complexity of haptic search tasks and makes it even more interesting that humans can do this with high accuracy and seemingly little effort.
A lot of research was done to investigate human behavior in such tasks, but to this day it was merely investigated what the approaches are to implement such behavior in technical systems such as robots. This work wants to consider two questions with the data recorded in a haptic search experiment with a multimodal glove worn by participants: Is it possible to inspect these pop-out effects based on tactile data and pose-information as well as the question of the applicability of these data in machine learning tasks.

1.2 Goals

A first goal of this work was to provide a setting where haptic search tasks could be performed by participants and be recorded with suitable hardware. In section 2 the haptic search experiment will be proposed and discussed starting by the experimental setup, the execution, hardware and the overall setting.
After having recorded the data a second goal aimed at generating a data set that was suitable for supervised learning tasks. These included postprocessing tha raw data

and generating labels for the individual trials. Section 3 describes the efforts that were made to reach this goal.
The last goal was to use the information collected with the data to perform machine learning tasks to investigate the differences between target- and distractor objects and measure the performance of the developed models. Section 4 will describe the different experiments and models that were used to address the question for applicability of recorded data from haptic search experiments performed by humans.

Chapter 2

Haptic Search Experiment

2.1 Haptic Search Experiment

2.1.1 Experimental Setup

The Modular Haptic Stimuli Board (MHSB) makes up the core part of the experiment. It is a setting with two wooden frames that hold stimuli objects. These objects are 3 x 3 cm big wooden blocks, which have a primitive three-dimensional shape on top of it or are just plane. The whole set consists of 360 blocks with 55 different shapes.
The first wooden frame can fit 25 objects and is used for learning a target object whereas the second frame has a capacity of 100 objects and is used for searching target objects. The stimuli are statically installed in the frames and not manipulable to allow a focus on just the search task itself (see Figure 2.1).

For this experiment, a subset of stimuli was chosen, consisting of 5 different shapes and plane ones (see Figure 2.2). The target consists of one object and is placed central in the small frame with the rest of the space consisting of plane stimuli. The big frame contains the rest of this subset, where each shape exist 4 to 5 times, including the target. The objects were distributed mostly equally and kept the same rotation throughout the experiment. Only the distribution and the target were changed with each trial.

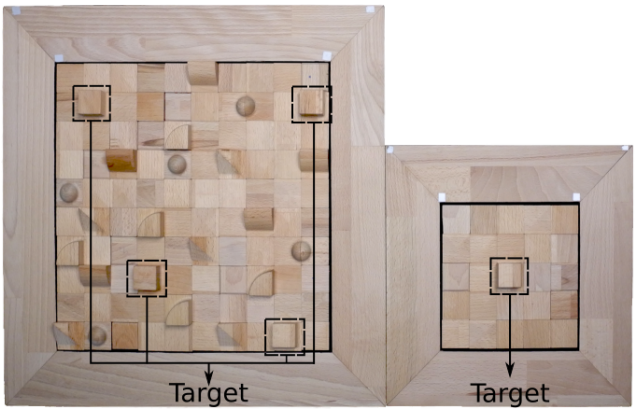

FIGURE 2.1: MHSB: on the right side for learning, on the left for search task. The marked regions show the target instances.

FIGURE 2.2: Stimuli objects used in the MHSB. From left: Wave, Sphere, Quarter, Box, Pyramid

2.1.2 Execution

For this experiment, 7 participants were invited and asked to solve a haptic search task while being blindfolded. The participants were 23 to 28 years old and included both genders. All participants were right-handed and have never seen the stimuli objects, so that during the task they never knew how the set of objects looked like and their perception was purely based on the haptic features.
Each participant performed on maximal 5 trials, where after each trial, the target was exchanged and the distribution of the stimuli on the big frame was changed. Before the beginning, there were 2 rehearsals to accustom the subjects to the setting. No participant had the same target twice or more and the task was done with just the right hand, while wearing a glove to record relevant data (See 2.2).

For the procedure, each participant was given a description of the task (See Appendix A). The task consisted of two parts.
The first task was to explore the target object on the small frame and remembering it just by its haptic features. When collected enough information about the target stimulus, the subject should proceed to the big frame and search for the learned target. The only goal in this part was to remember the approximate position of the target and not saying that it was found or pointing at it, so the recorded data would not contain pauses or pointing postures.
It was not necessary to find every target shape in the big frame, just as many as one could. The time was limited to 30 seconds to guarantee that the focus lies only on the salient features. An acoustic signal by the examiner determined the start- and endpoint of the experiment.
The second part of the experiment was to figure out if the subjects found the target object between the non-target objects, called distractors, and how well they could remember the approximate position on the frame. Again an acoustic signal determined start and end of the trial. For the second part, the subjects had just 10 seconds left to find the targets and point on them. The short period of time was set to prevent the subjects from exploring too much of the frame and focusing only on the smallest set of haptic features that were sufficient enough to differentiate between target and distractor.

2.2 Hardware

In this section the used hardware will be described as well as the overall setting that was used to record the data for the experiment.
There will be first a brief description of the glove that was used to capture tactile relevant and hand posture data followed by a description of the Vicon system to capture position data in a three-dimensional environment. At the end the implementation of the hardware into the experimental setting is explained.

2.2.1 Glove

A detailed explanation of the underlying technical properties and its implementation into the glove can be found in the work of Bianchi et. al. [2].

To record data for this experiment, a device was needed that would be able to capture the most relevant patterns underlying a haptic search task. These so called

exploratory features (EPs) describe the behavior of the hand during the exploration [8]. Furthermore a device for recording the tactile properties was needed.
The multi-modal sensing glove combines both of these features. On the bottom side of the glove 64 tactile cells are mounted, covering hand palm and fingers. These fabric-based sensors record local pressures with a frequency of 150 Hz. The top side consists of 18 bending sensors, used to capture the joint angles representing the hand pose with a frequency of 50 Hz.

2.2.2 Vicon

For capturing the position of the hand and the MHSB, the Vicon system was used [4]. It records motion data with a frequency of 200 Hz, using retroreflective markers that are tracked by infrared cameras.
Also included is a Basler camera, generating a top-down view for the experiment.

2.2.3 Setting

To record motion data from the subjects hand, 17 reflective markers has been placed on an extra glove that the participant wears atop of the multi-modal one. The markers were placed in a position as shown in Figure 2.3 to guarantee a good reconstruction of the finger and hand movements.
The most time-consuming part was to find a setting of the Vicon cameras that would capture the reflective markers continuously, making sure to minimize the occurrence of gaps. The result is shown in Figure 2.4 There were 14 Vicon cameras placed in a semicircle around the MHSB. The Basler camera was placed directly above the frames. As an addition, there were 2 cameras placed on the left and right side to record also the side-view of the experiment.
The glove was connected via USB and serial-port to a nearby computer. A second computer controlled the Vicon system. To simultanetly start the recording, a synchronising tool called MSS was used (See 3.2).

FIGURE 2.3: Glove with 17 retroreflective markers for tracking the hand movement with Vicon

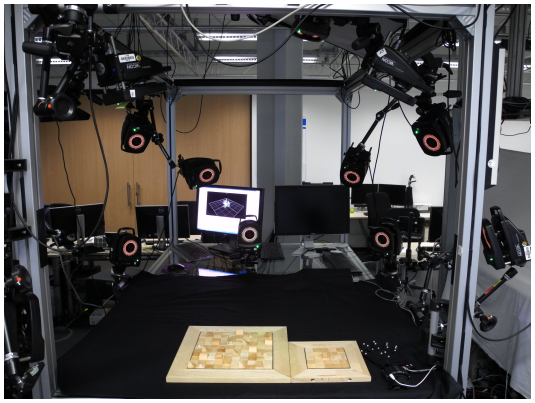

FIGURE 2.4: Experimental setting with MHSB, glove and Vicon cameras

Chapter 3

Data Generation and Analysis

This chapter addresses the methods and efforts to tackle the task of data generation as well as labeling the huge amount of data that was recorded.
It was the most time-consuming part of this work, since it involved a lot of post-processing and data cleansing work that was necessary due to the multi-modality of the recording devices and capturing data with different frequencies with various data formats which also were partly unsynchronized. Furthermore methods are explained that were used to label the generated data mostly automatically, based on position data of the hand and the MHSB, as well as a representation of the distribution of the stimuli objects on the frames.

3.1 Data Structure and Requirements

The data in this experiment was recorded with multiple devices, including the Vicon system, the 2 parts of the glove as well as 3 cameras, generating side- and top-views. To be able to train a classifier with supervised learning, there were a number of requirements to the data:

1. Simultaneous data acquisition
    - Capturing all devices at the same time will facilitate upcoming processing steps
2. Postprocessing raw data
    - To be able to work with the data, raw data needs to be processed and all files need to be in the same format
3. Synchronizing the time-series
    - Delays in the data acquisition and different device frequencies make this step necessary
4. Generating the labels
    - For supervised learning, the whole dataset needs to be labeled

3.2 Recording

To record the data of all devices preferably at the same time and with giving just one start signal, a tool called Multiple Start Synchronizer (MSS) was used. MSS sends a trigger signal to all registered devices which makes them start and stop capturing data.

The Vicon and Basler camera data were captured directly within the Vicon Nexus program. For the glove, data was recorded as rosbag consisting of two topics for each part of the glove. Side-view camera images were captured directly as image files.

Despite using MSS, there were still delays among the different devices that had to be synchronized separately.

3.3 Postprocessing Vicon Data

The first step in the pipeline was to postprocess the Vicon data. In this procedure, a three-dimensional hand model with marker positions was fitted to an image of the subjects hand (See Figure 3.1, left). This model was then used to reconstruct the hand movement during the experiment with an autotracker tool [5] to approximate marker positions that occurred during gaps in the recording when no camera captured a marker (See Figure 3.1, right).

The resulting file contains a time-series of the x-,y- and z-position of each marker. Furthermore a file with the joint-angles was generated.

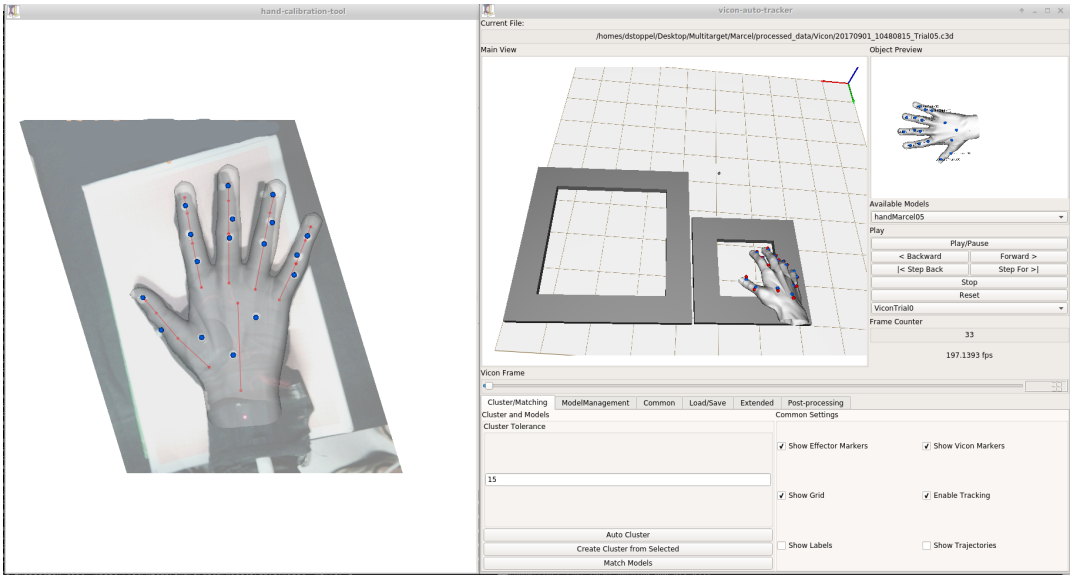

FIGURE 3.1: Fitting a hand model (left) and using it in the autotracker to fill gaps (right)

3.4 Generating Labels

This section will describe the methodology that was used to generate labels for the recorded data. The challenge was to write a program, that will do most of the work automatically and handle the huge amount of data generated by this experiment. With 7 subjects participating in up to 5 trials each, and a time series containing between 5000 and 7000 data points for each trial, a manual labeling of the data would be too time-consuming. Also having to cope with unsynchronized data due to delays between the modalities would make this task hard to tackle without proper preprocessing. The solution was a program that used the trajectories of the Vicon

data to extract objects that were explored during the search experiment and to label them appropriately. The exact procedure is described in the subsections below. It can be summarized to three mandatory steps:

1. Synchronizing data from Vicon and glove to allocate positions to tactile data
2. Building a representation of the experimental setting to recreate it in sense of the hand trajectories and object distribution
3. Generating labels by replaying these trajectories and constructing a vector containing the explored objects at each timestep

3.4.1 Synchronizing Glove and Vicon Data

A problem that occurred during the acquisition was the delay between starting the Vicon system and the glove recording. Although sending a trigger signal to both systems at the same time, the glove started capturing data approximately 3 to 5 seconds later. Additionally the beginning of the Vicon data had to be cut by 100 to 1000 frames for postprocessing reasons. Fitting the three-dimensional model was only successful if the markers of the first frames had a nearly perfect plane position. As a consequence, an offset had to be defined pointing to the beginning of the Vicon time series because the data only contains a timestamp describing the beginning of the recording. On the other hand the recorded rosbags from the glove came with a timestamp for each sample.
Since the frequency of the tactile glove with 150 Hz is lower than the frequency of Vicon with 200 Hz, the trajectory data should be reduced to the length of the tactile glove.

Consider the two time series $V = \{v_t \mid t \in T_V\}$ and $G = \{g_t \mid t \in T_G\}$ describing the set for the Vicon data and glove data. The set of timestamps T_G was given for the tactile data and consisting of unix time values. For T_V the timestamps had to be calculated for each sample from the initial timestamp, the offset and the frequency.
To synchronize, a new time series $V' \subset V$ was defined with

$$V' = \{v_t \mid \forall g_{t_g} \in G \exists v_{t_v} \in V : t_v \geq t_g \wedge t_v < t_{g+1}, t_v \in T_V, t_g \in T_G\}$$

This new time series has now equal length to G and each time value from V' matches exactly one time value from the time series G.

3.4.2 Representing Glove and Objects

The core idea behind this program was to use the hands trajectories and approximated object positions to detect which object was covered by the hand during which time. Having the trajectory data given, the only thing that had to be done manually was the object distribution. For this, a representation of the board was generated in form of a matrix $B \in \mathbb{R}^{10\times10}$ for each trial. In this representation, $b_{11} \in \Omega = \{0, .., 5\}$ would be the top left object and from there rows and columns were generated accordingly where Ω is the set of labels. To decrease the number of false-positives, only explored objects were considered in this representation.
The next level was to represent this information in a coordinate system by generating polygons for each object in B to embody the MHSB. Since only the top corners of the boards were assembled with markers, the positions for respective corners of

the polygons had to be calculated based on this. First for each object $b_{ij} \in B$ a polygon $P_{ij} = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$ was created where each element describes the x- and y-position of a corner. In the second step, this polygon was represented as its center position $z = \begin{pmatrix} z_x \\ z_y \end{pmatrix}$. The result is a matrix $B' = \begin{pmatrix} z_{11} & \dots & z_{1n} \\ \vdots & \ddots & \vdots \\ z_{n1} & \dots & z_{nn} \end{pmatrix}$ that represents every object in B through a position.

The remaining problem was that the top left position of the bigger frame was used as a base to build our polygons using a step size corresponding to the edge length of the stimuli. As a result, the represented board was placed parallel to the x- and y-axis into the coordinate system. Because this did not match the real setting, the matrix B' had to be rotated. For this the actual angle α between the vector $\vec{t} = \mu_{tr} - \mu_{tl}$, where μ_{tl} is the mean position of the top left corner and μ_{tr} the mean position of the top right corner, and the x-axis had to be calculated to build a rotation matrix R_α. The angle α describes the angle that was needed to rotate the representation so that the orientation of B' matches the one in the real setting. The final matrix for the stimuli representation is $B'' = \begin{pmatrix} R_\alpha z_{11} & \dots & R_\alpha z_{1n} \\ \vdots & \ddots & \vdots \\ R_\alpha z_{n1} & \dots & R_\alpha z_{nn} \end{pmatrix}$.

With B'' a matrix was now given that could be used for assigning objects, or in this case their labels, to hand positions. But in order to do this, a representation for the hand had to be thought of. For now, the hand consisted of 17 trajectories t_i, one for each marker i.
A first approach was to use a convex hull H_c of the hand and check for each time step in V' if H_c contained points $p \in B''$. This idea was discarded quickly because the seize of the convex hull was to large, resulting in multiple possible objects p for each time step. Furthermore it was computationally costly since for every step the whole matrix B'' had to be checked for contained points. This led to an extension of B'' where a k-d-tree was build containing positions of all p such that it could be efficiently queried and search complexity could be reduced from $O(n^2)$ to $O(\log n)$.
The second approach simplified the representation by only using finger markers. Instead of the convex hull for the whole hand, the trajectories t_i were averaged for each finger, resulting in only 5 positions and a representation H'_c. Also there were no checks for points $p \in B''$ that were contained in the convex hull anymore, but rather finding the point p_i with minimum distance to the center of H'_c. These distances could be looked up now efficiently in the k-d-tree and there were no multiple possible objects for each time step but only one. This approach improved the performance greatly so that only a few points were still labeled falsely due to the size of H'_c.
The last approach was fine tuned by simplifying even more. Now only four fingers were used, the index-, middle- and ring finger together with the thumb. Observations showed, that the little finger wasn't used often during exploration. The result was a pyramid-like polygon that was precise enough to exclude false labels almost completely.

3.4.3 Finding Labels

Having now a representation of the glove and the objects in the MHSB, the remaining task was to bring it all together to find the labels for tactile data. In this subsection, the algorithm is explained that was written to label data almost automatically as well as further cleaning steps that were mandatory.

The algorithm[1] requires synchronized data, the representations of glove, objects and the target label. These were also part of the program, but were treated separately in the previous subsection since the focus here is the procedure on how to find labels.
The algorithm starts by iterating over all time steps in V'. For each iteration, first the hand representation is calculated by the current positions as explained previously. The center position of this polygon is then passed on to query the nearest object in the k-d-tree. Furthermore a mean position μ_z is calculated for the hands z-position. This will serve as validation condition to see if $\mu_z <= \delta$ with δ describing a threshold for the minimum height of the hand. It is approximately a bit above the boards height in three-dimensional space, since the representations are just in two-dimensional space and there is no information about the z-axis given. Additionally it validates whether the polygon center is inside one of the boards. If both conditions are true, the label is assigned for this time step.
For debugging purpose, the program also includes a simple visualization tool to follow the process that shows the representations of the objects and for each iteration the representation of the glove as well as the assigned label (See Figure 3.2).

Algorithm 1 Finding and assigning labels to a time series

Require: time series V' containing marker positions, k-d-tree T
begin
  Initialize $l = \{\}$ and threshold δ
  for every time step t in V' **do**
    $p \leftarrow generatePolygon(V'(t))$
    $label \leftarrow T.query(p.center)$
    $mean_z \leftarrow getMeanZ(V'(t))$
    if $mean_z \leq \delta$ and $IsInside(p.center)$ **then**
      $append(l, label)$
    else
      $append(l, 0)$ *//means no relevant object explored or hand outside*
    end if
  end for
  return l
end

A small addition was made after the first few observations. As a result of using only the center of the polygon, small noise in the position led to false labels when the center appeared to be closer to a neighbor object. To fix this problem, and additional parameter γ was added describing an attraction variable. If the same label occurs consecutively, meaning an object is explored for some duration, γ increases. When then the label changes, but the old one is still near, the algorithm will stick to the previous one while decreasing γ. This ensures to avoid gaps of false labels.

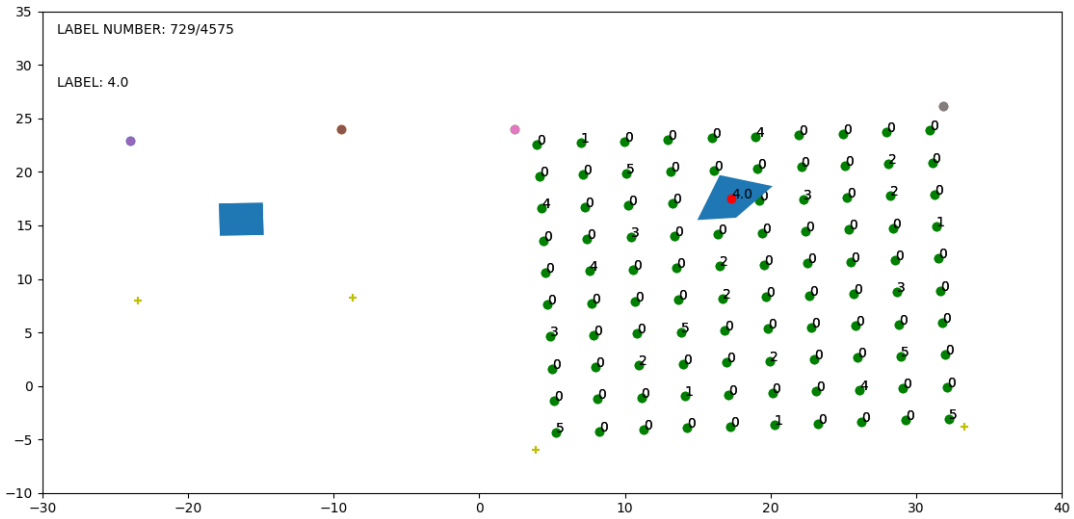

FIGURE 3.2: Auto-labeling visualization tool with the hand representation (blue polygon), the target object (blue square) and the search stimuli represented as their polygon center (green dots) and their respective labels

After generating the label vectors for the trials, almost no additional work had to be done manually. However if the data was too noisy, few gaps had to filled by hand with the help of the visualization tool or good guessing.

3.5 Analyzing the Data

For this experiment a total of seven subjects participated, each in up to five trials. After postprocessing the Vicon data, fitting a hand model and labeling, some problems were noticeable that led to an exclusion of trials from the final data set.
One problem was the losing of markers during an experiment. While most of these scenarios were detected and the trial repeated, a few cases went unnoticed until the postprocessing step. It was not possible to reconstruct the hand model with less markers anymore.
A second problem was to much noise that would act as ghost markers in later processing steps. The programs used were able to handle just a specific amount of noise, so that a few trials could not be labeled.

The final data set consists of data from **29** out of **35** trials and includes **137123** data points. Also included are the non relevant labels that describe cases where the hand was not exploring an object or outside the MHSB. In the table below the number of trials and data points for each participant A to G are listed:

Composition of the Dataset		
Participant	Trials	Data points
A	3	13867
B	3	13887
C	4	21230
D	5	25720
E	4	17895
F	5	20446
G	5	24078

(3.1)

An analysis of the sensor data revealed, that each participant had their own range of values, which most likely is correlated to their hand size and form. Smaller hands showed an slightly different area of activation for tactile data and also a significant different range for the joint angles. Due to this discovery, data from participants was treated separately rather than as one set.

Chapter 4

Evaluation

This chapter will present the evaluation and results of the goals that were set in 1.2. At first, approaches will be explained to evaluate the goals and related work will be presented. After this, the methods are proposed containing the preprocessing steps and the selection of the training and validation sets with respect to the different approaches. In the last step, results are presented and discussed.

4.1 Approaches

In this work there were three approaches to analyze the influence of object roles in the haptic search experiment. For all of them supervised machine learning was used resulting in three different classification problems:

1. **Classifying data into object categories:** a model was build to classify the five stimuli used in the experiment. At first the model was trained only on the data of objects when they were targets, and second on the data when they were distractors. The performance was measured and compared. The goal was to see if the data would be separable at all and to find a fitting model for it.
2. **Classifying a single object as either target or distractor:** based on the previous problem, same model was trained separately for each object to classify its data into a target role or a distractor role.
3. **Classifying unseen objects into roles:** in this problem, the model was trained on a single object as either target or distractor and tested on a different unseen object with the same role. The goal was to find out if it is possible to classify roles on an unseen object, which would give insight on the feature correlation between objects.

Combining the results of all these problems, an answer to the question whether humans explore same objects differently in a haptic search task depending on what the target is should be given. Furthermore an approach to explain the human efficiency could be made by the results. Instead of classifying all explored objects, they distinguish just between two classes, the target object they searched for and a distractor.

4.2 Related Work

This work combines both, object categorization based on their various characteristics and haptic search. Researchers have dealt with the specific task of object classification in previous studies. Since material and functional properties could not be

captured because the stimuli used were static and not deformable objects, previous work on shape based classification is perhaps the most related work to the proposed approaches.
Schneider et. al. [1] use touch sensors in a manipulation robots fingertips to gain low-resolution intensity images from multiple grasping interactions. They apply a bag-of-words approach and clustering techniques to categorize objects based on haptic feedback. Navarro et. al. presents an approach for haptic recognition and evaluation on multi-fingered robot hands based on extracting key features of tactile and kinesthetic data using clustering [9]. Faldella et. al [3] describes an approach to robotic haptic recognition using an unsupervised Kohonen self-organizing feature map for performing a match-to-sample classification of three-dimensional objects. Pezzementi et. al.views tactile sensor data as images and applies PCA techniques to identify principal components of identified features and clusters them as well as build per-class histograms as class characteristics [12]. Gorges et. al. [7] additionally includes passive joints in the tactile sensor system which could help to acquire more information for shape reconstructionn. They use Self-Organizing Maps for identifying haptic key features and a Bayes Classificator for classifying objects. Bhattacharjee et. al. [10] demonstrate a tactile sensor array covering a robot's forearm to generate haptic time series data during manipulation tasks. They use the processed and dimensionality reduced data to generate feature vectors and classify them with a k-nearest neighbor algorithm for object recognition.

Although it is not dealt with categorization of explicit shape features in the previously defined classification problems, the tactile data acquisition, preprocessing and feature extracting used in these works could also be applied for these approaches. Especially parts of the data sampling and preprocessing pipeline from Bhattacharjee et. al. [10] was found to be well applicable on the data recorded for this work.

4.3 Methods

In this section the pipeline is presented that was used for the classification problems of the evaluation. At first the preprocessing steps will be explained that will turn the raw data into feature vectors that can be used for training. Figure 4.1 depicts the complete experimental protocol. In the second part it will be described how the data sets for training and validation were generated.

4.3.1 Notation

A number of notations will be used in this work to describe the data sets that were used in the different evaluations.
Let $\Omega := \{(x_i, y_i)\}_{i=1,...,N}$ be the data set of one trial where N is the number of samples in the time series of this trial and the tuple (x_i, y_i) denotes the feature vector with the corresponding label at time i. Since every trial had one target object that the participant had to search for while the rest of the objects were distractors, the data set can be decomposed in $\Omega = \Omega_T \cup \Omega_D$. Here Ω_T refers to these (x, y) where $y = t$ is label of the target object $t \in \{1, ..., 5\}$. On the other hand Ω_D contains only the labels of distractor objects $y \in \{1, ..., 5\} \setminus t$. A complete set $\Omega_A = \Omega_1 \cup \ldots \cup \Omega_M$ describes the set containing all M trials of participant A.

4.3.2 Preprocessing and Feature Extraction

The first step in the pipeline was to apply a z-transformation for each sensor separately with $x'_j = \frac{x_j - \bar{x}_j}{\sigma_j}$, where x_j denotes the j-th component of all samples in the time series vector and $\bar{x}_j, \sigma_j$ are the mean and standard deviation. Standardizing the data to zero mean and unit variance was necessary since the different tactile cells and bending sensors all have various ranges based on the participants hand, search strategy and some noise which would significantly influence distance based classifiers.
Afterwards a time window was chosen to sample data from the time series at consistent intervals to reduce the amount of redundant data. The time series were recorded with 150Hz and 50Hz, resulting in very close or even similar neighboring data points. With the time window data points were picked that had a predefined time distance to the previously collected sample. In this work, a sampling rate of 10Hz proved itself reliable.
The next step was to extract only relevant samples dependent on the classification problem. Since one time series represents the data of a whole trial, just these data points had to be extracted that belong to specific objects, namely the ones to investigate. This had to be done individually for every approach. Only the data points that included no information at all, e.g. when the hand was in the air or outside of the MHSB, were discarded for all approaches.
In the last step, the extracted data was concatenated and a low dimensional representation of the data was computed using a feature extraction method. To find an optimal representation, multiple methods were applied in the first experiment and the one resulting in best performance of the model was chosen. These methods were principal component analysis (PCA) and autoencoder (AE) as well as handcrafted features like only finger sensors, the sum of the sensors on each finger and the maximum value of the sensors for each finger.

4.3.3 Training and Validation Methods

For training of the feature vectors and the different extraction methods, a multilayer perceptron (MLP) was applied on the data sets for the classification problems. The first experiment should serve as a baseline to see how well the data can be separated and which feature extraction would prove itself suitable. The MLP consisted of two hidden layers, was using an rectified unit activation function and the lbfgs method as an solver for weight optimization which comes from the family of quasi-Newton methods.

A problem that occurred when recording a trial in a single run was that the whole exploration was saved in a sole data frame. This is why the extraction step in the pipeline was necessary to generate data sets suited for the classification experiment. Training was done for each participant separately and scores were averaged since high variances yielded bad results on data sets covering all participants. For the different approaches the following data was extracted to build a training and validation set:

1. **Classifying data into object categories:** In fact this experiment included two sets of data. The first one was $\Omega_T = \Omega_{T_1} \cup \ldots \cup \Omega_{T_N}$ which was only the data of target objects of all N trials of a participant and respectively $\Omega_D = \Omega_{D_1} \cup \ldots \cup \Omega_{D_N}$. The model was trained on Ω_T and Ω_D independently resulting in two five-class classification problem. Comparing the performance of these

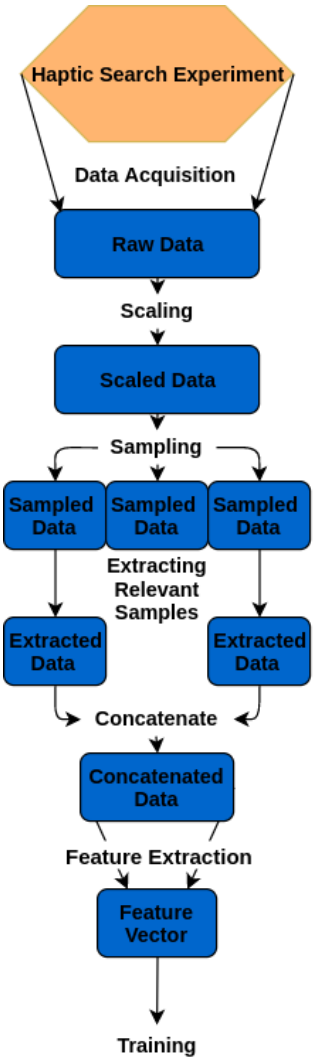

FIGURE 4.1: Schematic representation of the complete pipeline

sets on the models should show some insight in the information these data carries.

2. **Classifying a single object as either target or distractor:** Here a model M_i was trained on a set $\Omega_i = \Omega_{T_i} \cup \Omega'_{D_{j_1}} \cup \ldots \cup \Omega'_{D_{j_{N-1}}}$ where i describes one trial, Ω_{T_i} the target data of this trial and the $\Omega'_{D_{j_n}}$ describes a special case of the distractor data. Here the data for the same object that was target in trial i was extracted, but out of all other $N-1$ trials $j_n \neq i$. Hence Ω_i contains just data of one object but in the role as target and distractor. M_i was trained for all $i = 1, ..., N$ trials and averaged to see how well the role of one object can be classified.

3. **Classifying unseen objects into roles:** For this experiment multiple sets were created and evaluated separately. The goal was to see if roles can be classified when tested on an unseen object. For the first scenario a model was trained on data $\Omega_{train} = \Omega_{T_{train}} \cup \Omega_{D_i}$ where $\Omega_{T_{train}}$ was labeled as 1 for all target objects and Ω_{D_i} as 0 for one distractor object i. The evaluation was then done on a set $\Omega_{test} = \Omega_{T_{test}} \cup \Omega_{D_{j \neq i}}$ containing the rest of the target data and one unseen distractor object j. The second scenario followed the same principles

but instead of this time it was trained on all distractor objects and one target and tested on another unseen target object.

Having generated training sets for the experiments, what was left over were suitable validation sets to test the models for generalization on unseen data. Due to the complex procedure of generating the training sets through cutting the time series for relevant objects and concatenating them back over multiple trials, some problems appeared when it came to splitting the sets for validation purpose.
Sampling random data points for the test set yielded almost no errors in the evaluation. Since this seemed unrealistic it was found that this was not a good way to generalize on unseen data points because even after using a time window for sampling the unprocessed data, neighboring samples were still close to each other. Also they came from exploring the same object, so this data was basically not really unseen.
Another approach was to use cross-validation to make sure that blocks of data that was unseen were held out for validation. However, since the data was concatenated over multiple trials this led to unseen blocks that contained whole trials which resulted in high error rates.
The solution was to zip the data for all trials as shown in Figure 4.2. Splitting the sets of each trials into equally sized blocks and concatenating the sets blockwise resulted in an arrangement on which cross-validation could be applied. When dividing this set again into blocks and leaving one out, as it is done by cross-validation, each block would contain data from each trial. With this procedure the generalization could be tested. Splitting the trials in blocks of five and using a five-fold cross-validation on the resulting set was most suitable for the data in this work.

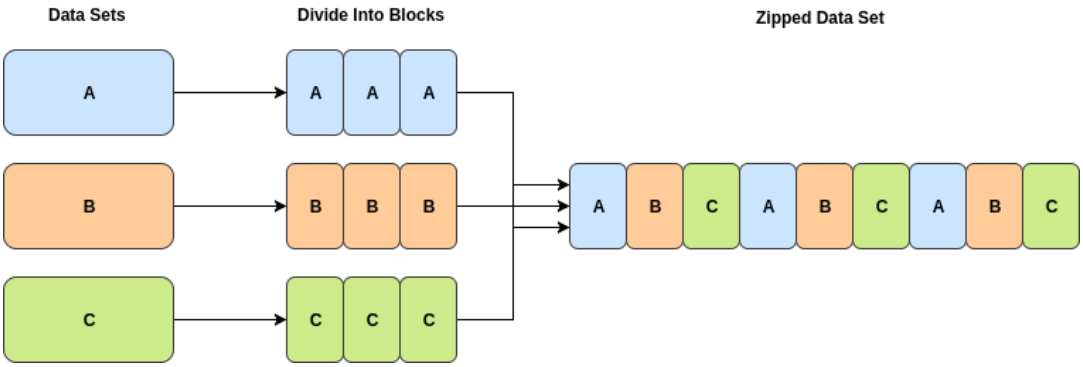

FIGURE 4.2: Schematic explanation of the zipping procedure to generate train and validation sets

Method	Specification
PCA	20 principal components
Autoencoder	2 encoding layers (1 hidden), 20 features, Optimization=Adadelta, Loss=MSE
Finger Values	Only finger sensors used (41 features)
Sum Of Fingers	Sum of sensors for each finger (5 features)
Max Of Fingers	Max value for each finger (5 features)

TABLE 4.1: Specifications for the used feature extraction methods

	Box	Sphere	Pyramid	Quarter	Wave
Box	43.4 (68.2)	12.2 (5.2)	8.3 (7.8)	13.7 (5.8)	22.1 (13.2)
Sphere	9.0 (6.1)	37.0 (62.7)	34.7 (11.2)	12.4 (4.4)	6.3 (3.9)
Pyramid	9.0 (4.7)	7.9 (9.0)	36.1 (56.0)	3.3 (4.9)	7.4 (6.2)
Quarter	22.3 (10.1)	20.6 (11.9)	20.8 (14.7)	64.1 (81.0)	6.3 (6.2)
Wave	16.3 (10.8)	22.2 (11.2)	0.0 (10.3)	6.5 (4.0)	57.9 (70.5)

TABLE 4.2: Confusion matrix with values in percent for the target data set and in brackets for the distractor data set

4.4 Results and Discussion

4.4.1 Classifying Data into Object Categories

For this experiment the MLP was first trained and evaluated on target data and then on the set containing distractor data. In both cases the training was done individually for each participant and the final score was then calculated by the average over all participants. Both variants were trained multiple times using different feature extraction methods. Each method was evaluated for the tactile data only, the merged set with joint angles concatenated with the tactile data and the joint angles alone.
The results for the model that was trained on target objects only is shown in Figure 4.4. Figure 4.5 shows the same experiment but this time trained on the distractor objects. Table 4.1 shows the used specifications for the different feature extraction methods. With the PCA the goal was to find a linear transformation for the data whereas the autoencoder with two encoding layers should find a non-linear representation. The other three methods used handcrafted features which were inspired by examining search strategies of participants and finding that most sensory activity can be reduced to the finger sensors. For the joint data no feature extraction method was used
The outcome shows that a training on only the tactile modality yielded best results on both, the target and distractor case. Also noticeable is that when using the merged modalities a significantly worse performance on the target data can be seen compared to the joint modality while on the distractor set the performance was almost identically between these two. In general, the model performed better on the distractor data which could be due to higher number of data points available.
Table 4.2 shows the confusion matrix for both data sets of this experiment. It appears that the pyramid stimuli has the highest number of confusions relatively for both data sets. Interesting is, that the sphere has the most confusions with the quarter and wave stimuli, which are the only objects besides the sphere that have also rounded features. This brings up the assumption that similar objects are more difficult to separate based on their tactile pattern.

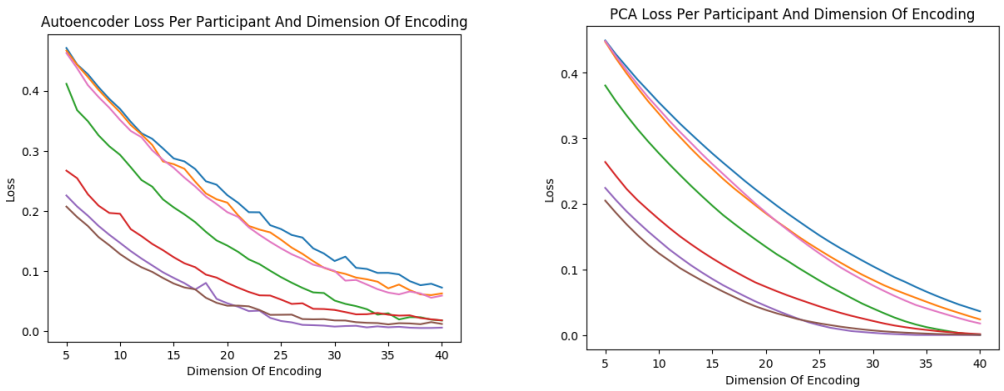

FIGURE 4.3: Reconstruction loss calculated with MSE for each participant. On the left side for the autoencoder and on the right side for the PCA method

Even thought five different feature methods were tested, none showed considerable better results.
An investigation of the reconstruction loss for the autoencoder and PCA method (Figure 4.3) revealed, that not only the performance of the respective model is similar, but also their loss behavior. It seems that the aspect of linearity or non-linearity in dimensionality reduction does not have significant influence on the reconstruction ability and model performance. Furthermore it can be seen, that there is a high variance for different participants in the loss function.
For the following experiments the choice for the feature extraction method fell in favor for taking only the finger sensors since they showed the most stable results. Also for further investigations the evaluation will be reduced to the tactile modality as there could no improvement be seen for the merged modality or only the joint angles.

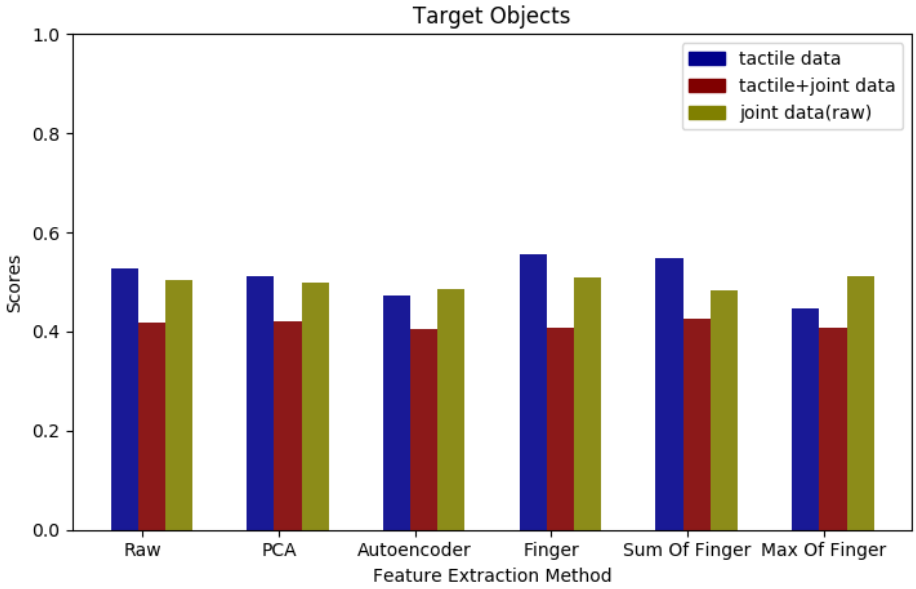

FIGURE 4.4: Scores of the MLP trained on target data with various feature extraction methods

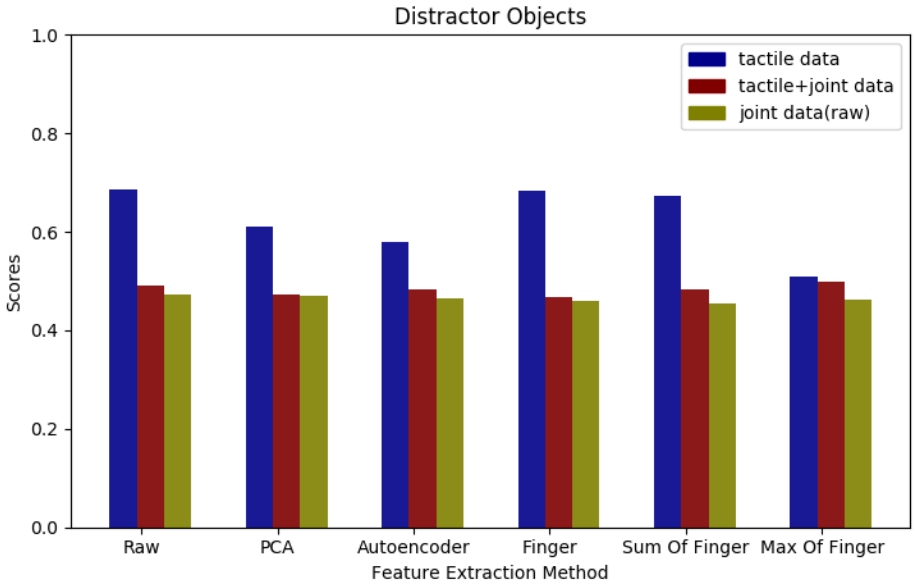

FIGURE 4.5: Scores of the MLP trained on distractor data with various feature extraction methods

4.4.2 Classifying a Single Object as Either Target or Distractor

This problem is an extension to the previous discussed one. It was found a compelling difference exist in the data regarding the role of an object which can be either target or distractor. For further examinations this problem aimed at finding out whether a classifier can find the role of an object based on the generated tactile data from the exploration. The data was trained with the multilayer perceptron using only finger sensors as features and evaluated with a five-fold cross-validation.
Figure 4.6 shows the result of classifying single objects as either target or distractor during the experiment for each stimuli. The accuracy score is composed of the mean scores of the respective objects trained separately on the data for each participant. The outcome shows high scores for all object types on a similar level. Also it can be seen that the pyramid stimuli shows the least accuracy which can be correlated to the confusion matrix (Table 4.2) that showed that this object has the least certainty of being correct classified in both, target and distractor data. However, an accuracy of almost 70% is still achieved.
These information proves that the glove can capture the crucial features that determine the objects role in the haptic search and it can be classified with reliable results. In the following experiment it should be further investigated how these roles correlate not only within a single class but on the whole class spectrum.

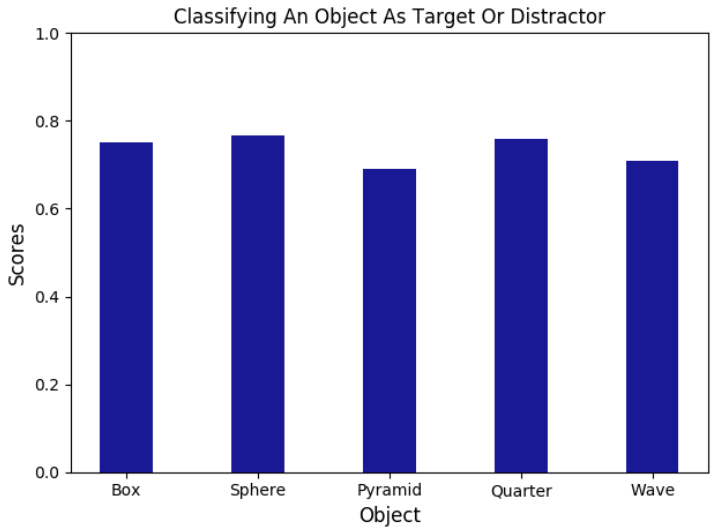

FIGURE 4.6: Mean scores of MLP trained on data of one object as target and distractor to classify the roles. Only tactile data was used together with the finger sensors as features.

4.4.3 Classifying unseen Objects into Roles

The aim of the last experiment is to find out if it is not only possible to distinguish the roles of one object class, but to see if this will work throughout classes of stimuli objects. For this approach a single object was learned as either target or distractor and tested on a set containing a different unseen object with the same role. Also included in the training and testing was the remaining data with the opposite role of the trained and tested objects. The model should then distinguish between target and distractor data for an unseen object. This will give some insight in the features that are specific for a role and if they are correlated.

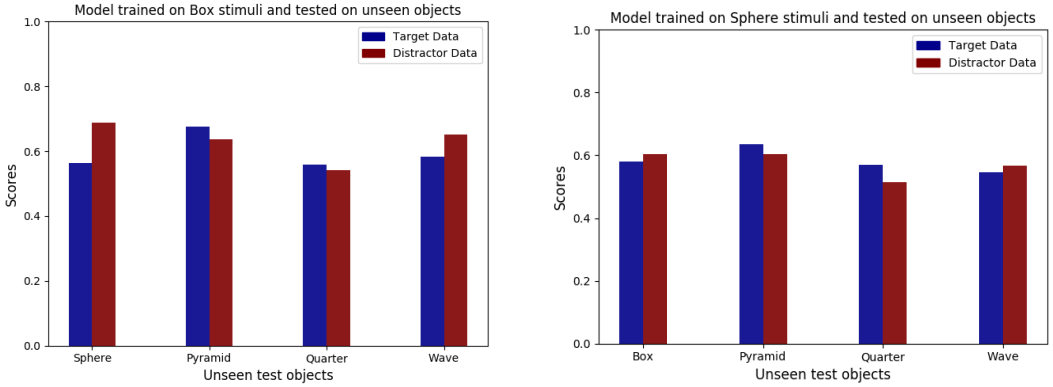

FIGURE 4.7: On the left side a MLP was trained on the Box stimuli, once as target and distractor and tested on the remaining objects with the respectively roles. The right side shows the same procedure for the Sphere stimuli.

Figure 4.7 shows two cases of this approach. Again, a MLP was trained on solely the tactile modality using the finger sensors as features. The training was done for each subject separately and the scores were averaged. On the left side the Box stimuli was trained once as target and once as distractor and it was tested for one of the

remaining stimuli each if it can distinguish between the roles. On the right side the same was done for the Sphere stimuli.
The outcome shows that for a model trained only on the Box stimuli, the Pyramid object can be classified with the highest certainty for the target and distractor case. This reinforces the assumption above since this object is the only stimuli besides the Box that has remarkable edges. But still the difference between the objects is small as it is also for the case where the model was trained on the Sphere stimuli. The second case displays even smaller differences. This could be because the Sphere object differs greatly from the other ones which means there are few to none similar features with the other objects.
All in all the results don't proof the assumption that there are role specific features since the scores are in a modest range and more testing would be mandatory. However, the scores are better than random which means that the MLP learns some regularity in the data to be able to achieve such scores.

Chapter 5

Conclusion

In this work three classification problems were proposed to investigate the influence of target and distractor roles using haptic information obtained from a multimodal sensing glove in a haptic search experiment. The data was preprocessed and a low-dimensional representation of the features was calculated using principal component analysis. The developed models were able to classify stimuli into one of five shapes: box, sphere, pyramid, quarter and wave. Furthermore it could be seen that this model performed much better when it was trained for objects that were target rather than distractors.
For the second investigation a random forest classifier was selected as winner model and performed a classification on the single objects and their roles. The results showed that the model can distinguish the role an object had in the scenario with high reliability. This proved on the one side that one can capture the differences with the multimodal glove and on the other hand brought up an assumption. This was that humans don't identify a distractor as an object itself, but rather as a class containing features that are not corresponding to the target. This is where it is assumed that efficiency comes from since just a small set of features are explored that are sufficient enough to tell the difference between the target and a distractor class but with no specific object knowledge. A further evidence was the result of the first experiment where it was barely possible to distinguish between the distractor objects.
The last investigation should corroborate this theory by showing that the model can separate all distractor objects as one class from the targets. As the outcome showed, it was possible to a certain degree to classify these classes.

Further inspections could now be to investigate the exact differences in the features that declare the same object as target or distractor. Also the data could be improved by new experiments where the hand size of the participates will be taken into consideration since high variances were measured for the experiments. Having more homogeneous test conditions will most likely result in more similar data.
A different direction to investigate in could be to test whether the model could find the target in an online setting faster than the human performing this experiment.

The results of this work combined with the further inspections could serve as valuable guidelines for experiments that use robots with tactile sensing ability to perform haptic search tasks to make them more efficient.

Appendix A

Experimental Instructions

JULIAN NOWAINSKI, UNIVERSITÄT BIELEFELD

Haptisches Explorationsexperiment

Versuchsanweisung

1 VERSUCHSBESCHREIBUNG

Bei diesem Versuch müssen Sie mit verbundenen Augen und einem Handschuh eine haptische Suchaufgabe lösen. Es werden Daten sowohl vom Handschuh als auch vom Vicon-System (Tracking von 3D Positionsdaten) aufgenommen sowie von mehreren Kameras (zwei seitlich und eine oben). Das Experiment besteht aus 2 Teilen.

Der **erste Teil** der Suchaufgabe beinhaltet das Lernen und Finden eines Zielobjektes in einem Holzbrett, welches aus Quadern verschiedener Formen besteht (das Lernen findet in einem seperatem Brett statt). Das Zielobjekt ist mehrfach darin enthalten. Sie müssen nicht alle finden, aber so viele Sie können. Es wird ein Zeitlimit von **30 Sekunden** geben.
Sie sollen nicht sagen, wo das Objekt ist, oder ob Sie es soeben ertastet haben. Merken Sie sich lediglich die ungefähre Position.
Im **zweiten Teil** müssen Sie die gefundenen Objekte aus dem Gedächtnis rufen (immer noch mit verbundenen Augen) und auf die Zielobjekte zeigen. Abtasten ist erlaubt, jedoch beachten Sie das geringe Zeitlimit von nur noch **10 Sekunden**

2 VERSUCHSDURCHFÜHRUNG

Insgesamt wird es 2 Durchläufe geben. Vor diesen gibt es einen Probedurchlauf. Die Dauer beträgt ca. eine Stunde

1. Desinfizieren Sie sich die Hände und ziehen Sie den Handschuh an (Hilfe wird geboten)
2. Als nächstes wird Ihnen eine Augenbinde aufgesetzt
3. Legen sie Ihre Hand auf die Startposition(rechts neben dem kleinen Brett) und warten Sie auf den Startbefehl. Hauen sie mit der flachen Hand (leicht) auf den Tisch
4. Ertasten und merken Sie sich das Objekt auf dem kleinen Brett und suchen Sie es auf dem großen. Sie haben insgesamt **30 Sekunden**
5. Warten Sie nach Ablauf der Zeit auf den Startbefehl für den zweiten Teil
6. Zeigen Sie auf die Zielobjekte (3 mal drauf tippen). Sie haben **10 Sekunden**

Bibliography

[1] J. Sturm A. Schneider and C. Stachniss. "Object Identification with Tactile Sensors using Bag-of-Features". In: *Proceedings of International Conference on Intelligent Robots and Systems (IROS)* (2009), pp. 243–248.

[2] Matteo Bianchi et al. "A Multi-Modal Sensing Glove for Human Manual-Interaction Studies". In: *Electronics* 5.3 (2016). DOI: 10.3390/electronics5030042.

[3] D. Passeri E. Faldella B. Fringuelli and L. Rosi. "A Neural Approach to Robotic Haptic Recognition of 3-D Objects Based on Kohonen Self-Organizing Feature Map". In: *IEEE Transactions on Industrial Electronics, vol. 44* (1997), pp. 267–269.

[4] *Information available at:* URL: http://www.vicon.com/.

[5] M. Schröder M. Botsch H. Ritter J. Maycock T. Rohling. "Fully automatic optical motion tracking using an inverse kinematics approach". In: *IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)* (2015).

[6] W. Bergmann Tiest M. Plaisier and A. Kappers. "Salient features in 3-D haptic shape perception". In: *Attention, Perception and Psychophysics* (2009).

[7] D. Goger N. Gorges S. E. Navarro and H. Worn. "Haptic Object Recognition using Passive Joints and Haptic Key Features". In: *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)* (2010), pp. 2349–2355.

[8] S. J. Lederman R. L. Klatzky and C. L. Reed. "There's more to touch than meets the eye: the salience of object attributes for haptics with and without vision". In: *Journal of Experimental Psychology* (1987).

[9] H. Wörn J.Schill T. Asfour S. E. Navarro N. Gorges and R. Dillmann. "Haptic object recognition for multi-fingered robot hands". In: *Haptics Symposium (HAPTICS)* (2012).

[10] C. C. Kemp T. Bhattacharjee J. M. Rehg. "Haptic Classification and Recognition of Objects Using a Tactile Sensing Forearm". In: *IEEE/RSJ International Conference on Intelligent Robots and Systems* (2012).

[11] Anne Treisman and Stephen Gormican. "Feature analysis in early vision: Evidence from search asymmetries." In: *Psychological Review* 95.1 (1988), 15–48. DOI: 10.1037//0033-295x.95.1.15.

[12] C. Reyda Z. Pezzementi E. Plaku and G. D. Hager. "Tactile Object Recognition from Appearance Information". In: *IEEE Transactions on Robotics, vol. 27* (2011), pp. 473–486.