Tactile Sensing for Manipulation

Verwendung taktiler Daten für Manipulation
Bachelor-Thesis von Nikolas Alexander Huhnstock aus Bad Nauheim
Oktober 2014

TECHNISCHE UNIVERSITÄT DARMSTADT

Tactile Sensing for Manipulation
Verwendung taktiler Daten für Manipulation

Vorgelegte Bachelor-Thesis von Nikolas Alexander Huhnstock aus Bad Nauheim

1. Gutachten: Prof. Dr. Jan Peters
2. Gutachten: M. Sc. Herke van Hoof
3. Gutachten: M. Sc. Filipe Veiga

Tag der Einreichung:

Please cite this document with:
URN: urn:nbn:de:tuda-tuprints-38321
URL: http://tuprints.ulb.tu-darmstadt.de/id/eprint/3832

Erklärung zur Bachelor-Thesis

Hiermit versichere ich, die vorliegende Bachelor-Thesis ohne Hilfe Dritter und nur mit den angegebenen Quellen und Hilfsmitteln angefertigt zu haben. Alle Stellen, die aus Quellen entnommen wurden, sind als solche kenntlich gemacht. Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.
In der abgegebenen Thesis stimmen die schriftliche und elektronische Fassung überein.

Darmstadt, den 1. Oktober 2014

(Nikolas Alexander Huhnstock)

Thesis Statement

I herewith formally declare that I have written the submitted thesis independently. I did not use any outside support except for the quoted literature and other sources mentioned in the paper. I clearly marked and separately listed all of the literature and all of the other sources which I employed when producing this academic work, either literally or in content. This thesis has not been handed in or published before in the same or similar form.
In the submitted thesis the written copies and the electronic version are identical in content.

Darmstadt, October 1, 2014

(Nikolas Alexander Huhnstock)

Abstract

In this thesis we want to make use of the rich tactile feedback offered by the human inspired fingertip sensor named BioTac. Our goal is to determine the forces and torques occurring between the sensor and an object during contact. We extract these features using the BioTac to enable more complex manipulation tasks. We are using a Gaussian Process (GP) to predict the arising forces and torques. A GP is well adaptable to training data when the hyper-parameters are set appropriately. Therefore we employ hyper-parameter optimisation based on the maximisation of the marginal loglikelihood. Additionally to the BioTac we use a force-torque sensor which provides the reference data for training and testing. We compare the performance of our approach to a Neural Network implementation. To prove the applicability of our method we are performing two demonstrations. One is related to guarded movements where the task is to stop when an obstacle is reached. The second task is related to guided movements where the BioTac is used as a joystick for the robot. Both demonstrations indicate a broad field of application for our methods.

Zusammenfassung

In dieser Thesis wollen wir das taktile Feedback von einem, einer menschlichen Fingerspitze nachgebildeten, Sensor namens BioTac nutzen. Unser Ziel ist es Kräfte und Momente zu bestimmen, die zwischen dem Sensor und einem Objekt bei Kontakt auftreten. Dies wird durch die Anwendung eines Gaussian Process (GP) ermöglicht, da sich dieser bei korrekt gewählten Hyperparametern gut an verschiedene Trainingsdaten anpassen lässt. Zur Sicherstellung verwenden wir Hyperparameter-Optimisation, welche auf der Maximierung des 'marginal log likelihood' basiert. Neben dem BioTac Sensor verwenden wir einen Kraft-Momenten-Sensor, welcher uns mit Referenzdaten zum Trainieren und Testen unseres GPs versorgt. Wir vergleichen die Performanz unseres Ansatzes mit der eines Neural Networks. Um die Anwendbarkeit unserer Methode zu zeigen, haben wir zwei Demonstrationen durchgeführt. Die erste Demonstration zeigt in Anlehnung an 'guarded movements', dass die Bewegung angehalten wird, sobald ein Hindernis erreicht wird. Die Zweite, angelehnt an 'guided movements', zeigt die Verwendung des BioTac als Joystick für den Roboter. Beide Demonstrationen zeigen einen umfangreichen Anwendungsbereich unserer Methode auf.

Acknowledgments

First of all I want to thank my two supervisors M. Sc. Herke van Hoof and M. Sc. Filipe Veiga for giving me great support throughout this thesis. They were always present when I had questions or needed a hint. Especially I want thank for all the time each of them did invest helping me to setup and fix system in the lab.

Also I want to thank Prof. Dr. Jan Peters for giving me the opportunity to work in a highly interesting field of research and this actual research topic.

Most importantly, I want to thank my family, none of this could have happened without the backing of my wife. I want to thank you for for all the sacrifices that you have made on my behalf. Your support and the certainty that our daughter is in best hands allowed me to focus at writing, especially during the final period of this thesis.

Contents

Figures and Tables

List of Figures

List of Tables

Abbreviations, Symbols and Operators

List of Abbreviations

Notation	Description
DOF	Degrees-of-Freedom
GP	Gaussian Process
i.i.d.	independently and identically distributed
NN	Neural Network
RMSE	root-mean-squared error
w.r.t.	with respect to

List of Symbols

Notation	Description
β	precision of the noise
$\mathbf{C}$	Covariance matrix
$\boldsymbol{\theta}$	vector of hyper-parameters
$\mathbf{I}$	Identity matrix
$\mathbf{K}$	Gram matrix
$\mathscr{L}$	the log marginal likelihood
π	the mathematical constant pi
$\mathbf{t}$	vector of training output data

$\mathbf{x}$	vector of test points
$\mathbf{y}$	vector of training input data

List of Operators

Notation	Description	Operator
det	the determinant of a matrix	$\lvert \bullet \rvert$
exp	the exponential function	$\exp(\bullet)$
N	the Gaussian distribution	$\mathrm{N}(\bullet)$
ln	the natural logarithm	$\ln(\bullet)$
Tr	the trace of a square matrix	$\mathrm{Tr}(\bullet)$

1 Introduction

It will be a major task for robots to be able to perform dexterous manipulation tasks if they are supposed to support a human in his daily live, whether as an independent system working beside him or as an prosthetic finger, hand or arm.

Since the time humans started to conquer earth one of the major abilities that differentiates us from most of the other creatures is the ability to build and use tools [1].

A vast amount of tools have been developed for all kind of tasks and hence nowadays we are surrounded by these tools. These tools make our life easier and if robots should act in our world and perform complex tasks, we conclude that it would be rational if they could use the tools that are already developed and almost always present. In theory it would be possible to equip the robot with several custom developed on-board tools to avoid using standard tools, but that would probably lead to a very limited usage of the robot.

During the usage of tools robots need to physically interact with them. The major field of object manipulation is dedicated to the manipulation of an object's position and orientation and this is also the kind of manipulations we wish to perform while using tools. When cutting a slice of bread we are just changing the relative location of the knife compared to the bread while maintaining the orientation to get a straight cut.

If we now think of more tools that we use daily we will come up with a mug, a cup, a door handle, a pen and several others. All of these are developed to be used by humans and it is not hard to come to the conclusion, that nearly all of then are made for interacting with hands. Thus robots will have to have hands human inspired hands to be able to use our tools.

Several human inspired robotic hands have already been developed [2], [3], [4], [5], but there is more needed than just a hand with fingers to perform dexterous manipulation tasks. Knowledge about the manipulation process is needed to control the required movements.

In manipulation tasks, the physical contact between the hand and the object is at the centre of focus. A contact situation can be described, for example, by the contact location on each of the objects/participants surfaces and the duration the contact is maintained. Furthermore there are properties that directly influence the objects state[1]. Examples of these properties are temperature difference, direction and amount of force, torque and vibrations. These properties describe the way the objects interact with another and so they also describe the way one object is manipulated.

In object manipulation the task is to manipulate these properties to reach a given goal state. In a tool usage task a goal state can be defined through a change in position and/or orientation. Changes on an objects position and orientation are performed through the appliance of an appropriate force and/or torque at a specific position. Therefore we need precise methods to determine the forces and torques during contact. To achieve this we need to equip the robot's fingers with sensors capable of providing us with feedback during contact.

The design and nimbleness of the human hand leads to strong size constraints. Furthermore collecting data at the contact location, the fingertip, enables us to collect data that sensors at joint side are unable to. Therefore we use a tactile sensor which is able to acquire data directly at the fingertip instead of a conventional force-torque sensor which is to big to use at every finger.

In this thesis we want to make use of the rich tactile feedback offered by the BioTac to determine the forces and torques occurring during object manipulation. The BioTac sensor is a human inspired fingertip sensor that provides a variety of sensory data [6].

We show a method capable of accurately mapping the BioTac data to forces and torques. To achieve this we use Gaussian Processes (GPs) because they easily adapt to different kind of input data, are well understood and are commonly used for this kind of applications.

1.1 Related Work

In general extracting forces from a tactile sensor exclusively has not been the subject of research/publications very often but it occurs in the majority of publications related to tactile based grasping [7], slip detection [8], compliance detection [9] or haptic adjectives determination [10]. For all these applications precise force determination is essential. Even in human diagnostics like tumor detection contact force determination from a tactile sensor can be applied and be of great benefit [11].

In consideration of the BioTacs very unique design, as described in section 3.1, it is hard to compare our work to work where flat arrayed sensors were used like in [8], where the authors additionally to the pressure sensor use an

[1] In this context the state of an object describes all properties of the object at a time.

accelerometer to mimic the human channels related with touch to improve pick and place tasks. In [7] Chitta et al. use the grasping force and the arising deformations to train a classifier which decides whether a container is filled or not. Ho et al interpret in [12] feedback of an flat pressure sensor as gray scale image and extract beside force, location and orientation also arising micro slips.

Thus we focus at publications involving the BioTac sensor. For example force extraction has been done by Su et al. [9] who use a transformation matrix and map the impedances measured by each electrode to a global frame by making use of the explicit location of each of the electrodes. In [13] Wettels et al. indicate, that a Gaussian mixture model is most effective at interpreting data from the BioTac but do not go into more detail about the implementation or results.

In a publication from 2011 Wettels and Leob state, that Neural Networks (NNs) were best suited to extract features from the BioTac [14]. They focus on three features: tri-axial forces, point of application of force and radius of curvature. Highly correlated to the topic of this thesis is the force vector extraction they describe and thus will be used as a comparison to our method.

Wettels and Leob mounted the BioTac sensor on a force plate to obtain correlated data from both sensors. They compared different dimensionality reduction methods which were applied on the input data before training the NN. Their results indicate, that when extracting tri-axial forces principle component analysis increased performance the most. We decided to not employ a dimensionality reduction method to prevent any loss of feedback obtained by the BioTac.

1.2 Outline

In the second chapter the used algorithm will be described mathematically. In the third chapter the sensors as well as the way data we collect the training and testing data will be decribed. After that the experiments and demonstrations we perform will be outlined. In chapter 4 the results of the experiments and demonstrations will be presented, a conclusion of this thesis and some proposals for future work will be given.

2 Theory

2.1 Gaussian Process

A Gaussian Process (GP), named after the German mathematician Carl Friedrich Gauss (1777-1855) [15], is a statistical method to approximate a function just known by some training samples. Given a specific function, called kernel function and a set of variables, called hyper-parameters, a GP describes a distribution over functions that may approximate the target function. Gaussian Processes are mainly based on the assumption of a Gaussian distribution over the input values and multivariate Gaussian distribution over the output values.

In the context of this thesis we use Gaussian Processes to solve a regression problem, which consists of estimating new target values out of a given set of input training data.

We begin with the assumption that our target value can be described, with the input data under a certain error or noise level.

$$t_n = y_n + \epsilon_n \tag{2.1}$$

where t_n is the target value, y_n are the input values of the real underlying function and ϵ_n is a normal Gaussian distributed noise with zero mean $N(0, \sigma^2)$. Furthermore this noise is independently and identically distributed (i.i.d.) over the single input values so that the probability of the target value becomes

$$p\left(t_n | y_n\right) = N\left(t_n | y_n, \beta^{-1}\right) \tag{2.2}$$

where β is called the precision of the noise.

When now putting all target values and all input values each into one vector $\mathbf{t}$ and $\mathbf{y}$. Considering that the Gaussian noise is i.i.d., we can describe the joint distribution of these as

$$p\left(\mathbf{t}|\mathbf{y}\right) = N\left(\mathbf{t}|\mathbf{y}, \beta^{-1}\mathbf{I}\right) \tag{2.3}$$

where $\mathbf{I}$ states the identity matrix.

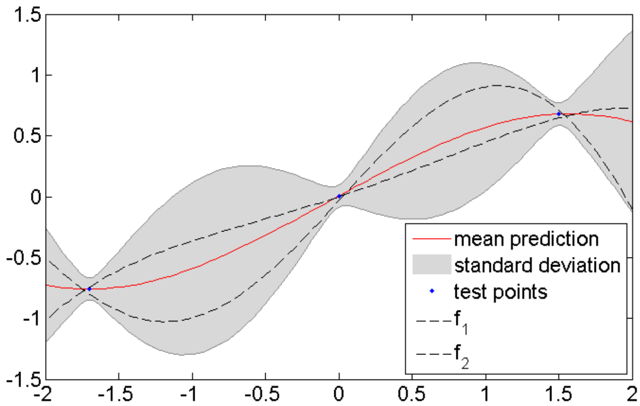


Figure 2.1: Plot of a prediction based on a Gaussian Process with three test points

The next step to get to the distribution $p(\mathbf{t})$ is to integrate over $\mathbf{y}$. To do this we need the marginal distribution $p(\mathbf{y})$ which by assumption is given by a zero mean Gaussian distribution

$$p\left(\mathbf{y}\right) = N\left(\mathbf{y}|\mathbf{0}, \mathbf{K}\right) \tag{2.4}$$

where $\mathbf{K}$ is called Gram matrix with elements $\mathbf{K}_{ij} = k(x_i, x_j)$. Where k is the kernel function and x_i and x_j are respectively the ith and jth input training point.

This leads to the marginal distribution of $\mathbf{t}$ which is given by

$$p(\mathbf{t}) = \int p(\mathbf{t}|\mathbf{y})\, p(\mathbf{y})\ d\mathbf{y} = N(\mathbf{t}|\mathbf{0}, \mathbf{C}) \tag{2.5}$$

where $\mathbf{C}$ is the covariance matrix with elements $\mathbf{C}_{ij} = k\left(x_i, x_j\right) + \beta^{-1}\delta_{ij}$.

Solving the problem of regression for a new target value t^* we need to calculate/evaluate the conditional distribution $p(t^*|\mathbf{t}, x^*, \mathbf{x})$. As shown above we take the joint distribution over $[\mathbf{t}; t^*]$ (which we will call $\mathbf{t}^*$). The distribution is given by

$$p(\mathbf{t}^*) = N(\mathbf{t}^*|\mathbf{0}, \mathbf{C}^*) \tag{2.6}$$

where $\mathbf{C}^*$ donates the corresponding covariance matrix, which is formed by applying the covariance function on the newly formed vector $\mathbf{t}^*$, equivalent to the creation of $\mathbf{C}$ defined earlier.

Taking a closer look at this covariance matrix $\mathbf{C}^*$ it is obvious that it is just an extension of $\mathbf{C}$ with one additionally row and column. We declare the partition of $\mathbf{C}^*$ as :

$$\mathbf{C}^* = \begin{bmatrix} \mathbf{C} & \mathbf{k} \\ \mathbf{k}^T & c \end{bmatrix} \tag{2.7}$$

where $\mathbf{k}$ is the vector with the elements defined by k(x,x*), x is the vector of training inputs, and c is the scalar formed by applying the covariance function on $\mathbf{t}^*$ and adding the inverted precision of the noise.

With the knowledge that the joint distribution over $\mathbf{t}^*$ is still a Gaussian distribution we are able to pose the mean and covariance of the conditional distribution $p(t^*|\mathbf{t}, x^*, \mathbf{x})$ as

$$\operatorname{mean}(\mathbf{t}^*) = \mathbf{k}^T\mathbf{C}^{-1}\mathbf{t} \tag{2.8}$$

$$\operatorname{var}(\mathbf{t}^*) = c - \mathbf{k}^T\mathbf{C}^{-1}\mathbf{k} \tag{2.9}$$

these values are the ones we aimed for and we now have calculated a prediction of the target value $\mathbf{t}^*$[16].

Figure 2.1 shows a plot of the mean (red) and the standard deviation (gray area) of a GP, with a Gaussian Kernel applied, based on three input points (blue). The plot illustrates that the standard deviation increases where no data is provided and decreases when near any given point. The grey dashed lines are two possible functions that lie within the standard deviation of the prediction.

2.2 Kernel Methods/Covariance Function

Kernels or covariance functions are symmetric functions of its arguments and take the form $k(x, x') = \phi(x)^T\phi(x')$, where ϕ can be any sort of basis function. The appropriate choice for a kernel is very important for the model of the GP, because different kernels show different correlations in the input data. There are several methods/rules for creating valid kernels which can be found in [16].

2.2.1 Gaussian Kernel

The Gaussian or Squared Exponential kernel is one of the most commonly used kernels and takes the form [15]:

$$k\left(x, x'\right) = \exp\left(-\frac{1}{2}||x - x'||^2\right) \tag{2.10}$$

and with added hyper-parameters:

$$k\left(x, x'\right) = \theta_1 \exp\left(-\frac{1}{2}\theta_2||x - x'||^2\right) \tag{2.11}$$

Hyper-parameters are used to improve the fit of the kernel to the training data. In the case of the Gaussian Kernel θ_1 is correlated to scaling and θ_2 allows to modify the width of the region of influence which effects the smoothness of the

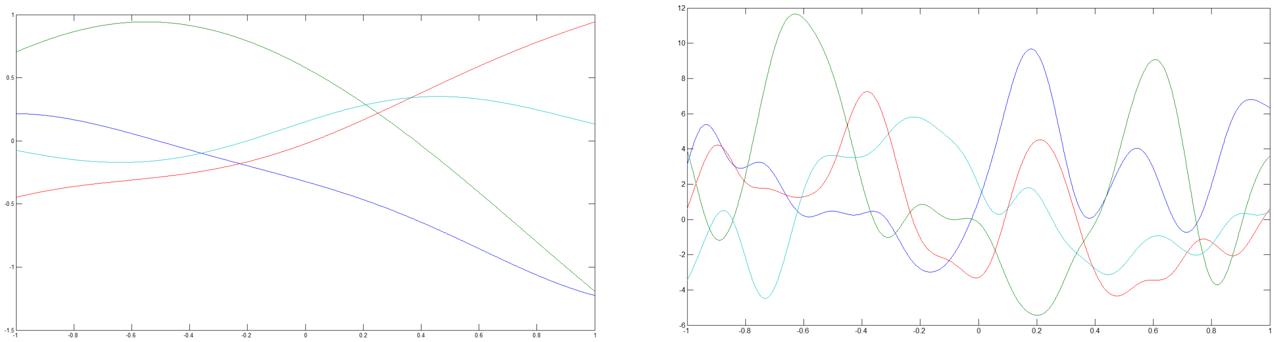


Figure 2.2: Influence of the hyper-parameters of a Gaussian Kernel

kernel. Figure 2.2 illustrates the influence of the hyper-parameters on the kernel function. The figure shows samples of two Gaussian Kernels with different values for the hyper-parameters side by side. In each plot 4 sample functions are drawn.

The value of θ_1 is reflected in the scaling of the vertical axis. In the left plot the values of the kernels are very small compared to the values of the functions in the right plot. This indicates that θ_1 of the functions shown in the left is smaller than θ_1 of the functions shown in the right plot. Also the influence of θ_2 can be easily shown in this figure. On the left side the functions are changing their values very slowly when moving along the horizontal compared to the right plot where the values are changing at a higher frequency. This indicates, that θ_2 on the left side is smaller than θ_2 of the right side.

As you can see, excluding the hyper-parameters, the Gaussian kernel just depends on the euclidean distance $||x-x'||^2$, makeing the Gaussian kernel a stationary kernel.

2.3 Hyper-parameter optimisation

As already mentioned hyper-parameters are used to fit covariance functions as good as possible to the training data. In this context we see β (the precision of the input data) also as a hyper-parameter in addition to the hyper-parameters from Section 2.2.1. To compare different sets of hyper-parameters we use the log marginal likelihood:

$$\mathscr{L}(\theta) = \ln p(\mathbf{t}|\theta) = -\frac{1}{2}\mathbf{t}^T\mathbf{C}_\theta{}^{-1}\mathbf{t} - \frac{1}{2}\ln|\mathbf{C}_\theta| - \frac{D}{2}\ln(2\pi) \tag{2.12}$$

In this equation θ denotes the vector of hyper-parameters $\theta = [\theta_1, \theta_2, \beta]$, the vector $\mathbf{t}$ denotes the training input data, $\mathbf{C}$ is the covariance matrix which is dependend of the hyper-parameters θ and $|\mathbf{C}|$ is its determinant. The parameter D designates the dimension of the training data.

Maximizing the fit is equivalent to the maximization of the log marginal likelihood. We are using a gradient based optimisation algorithm to find extrema, therefore we need the derivatives of the log marginal likelihood. The partial derivative with respect to (w.r.t.) the hyper-parameters is given by:

$$\frac{\partial \mathscr{L}(\theta)}{\partial \theta_j} = \frac{1}{2}\mathbf{t}^T\mathbf{C}^{-1}\frac{\partial \mathbf{C}}{\partial \theta_j}\mathbf{C}^{-1}\mathbf{t} - \frac{1}{2}Tr\left[\mathbf{C}^{-1}\frac{\partial \mathbf{C}}{\partial \theta_j}\right] \tag{2.13}$$

The partial derivatives of the covariance matrix, which is built applying a Gaussian kernel, w.r.t. the hyper-parameters are given below.

$$\frac{\partial \mathbf{C}}{\partial \theta_1} = \exp\left(-\frac{1}{2}\theta_2||x-x'||^2\right) \tag{2.14}$$

$$\frac{\partial \mathbf{C}}{\partial \theta_2} = -\frac{1}{2}\theta_2||x-x'||^2\theta_1\exp\left(-\frac{1}{2}\theta_2||x-x'||^2\right) \tag{2.15}$$

$$\frac{\partial \mathbf{C}}{\partial \beta} = I \tag{2.16}$$

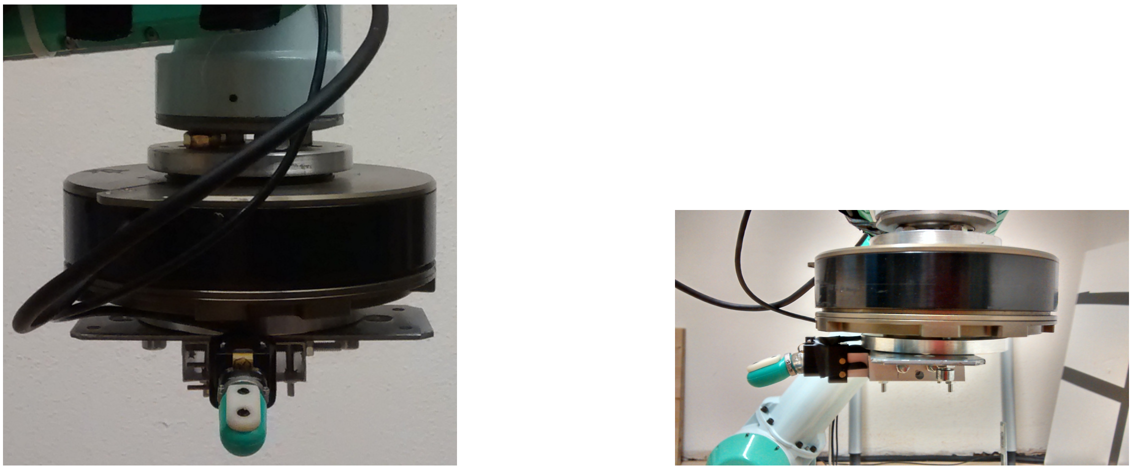
3 Experimental procedure

To train our Gaussian Process (GP) we need to acquire data. In this chapter we describe the data collection process and take a look at the recorded data. We introduce the sensors involved and how they are arranged. After that we introduce the method we compare our GP to and describe the experiments we have performed. Finally we describe the demonstrations we have implemented.

3.1 Setup and Sensors

The following images show the way the sensors are attached to each other and to the robot. As shown on figure 3.1 the BioTac is mounted in an 90 degree angle on the force torque sensor, which itself is directly mounted onto the robot.

We are taking care of the orientations and locations of the different frames arising in this setup by applying transformation matrices that transform forces from one frame to another. This is needed to enable us to compare the forces applied to the BioTac with forces arising at the force-torque sensor and to transform our predictions in the robots task space where the movements are performed.



(a) front view **(b)** side view

Figure 3.1: The arrangement of the sensors and the robot

3.1.1 BioTac Sensor

Figure 3.2: The SynTouch LLC BioTac Sensor [6]

The BioTac sensor, shown in figure 3.2, is a human inspired fingertip sensor which is built to be used on robotic or prosthetic hands. It has a soft silicone skin which covers the core. The space between the core and the skin is filled with a conductive liquid which gives a compliance similar to the human fingertip.

The BioTac has 19 electrodes which are placed at the rigid core and measure the impedance of the liquid. When force is applied, the flexible skin will deform and distort the fluid underneath resulting in a bulge on the other sides of the sensor, thus all electrodes respond to the applied force. The BioTac also measures the overall pressure and the vibration of the liquid. Furthermore the sensor provides data about the temperature and the temperature flow [14].

3.1.2 Force-Torque Sensor

Figure 3.3: The Schunk FTCL force-torque Sensor [17]

The force-torque sensor type FTCL shown in figure 3.3 is produced by Schunk and measures forces and torques in all directions. The sensor is able to collect data at a rate of 1 kHz [17]. We use the data provided by the sensor for different purposes: During the training phase of our GP we are using the data as training output samples. During the testing phase and during the experiments we compare our predictions to reference signal provided by the sensor.

3.1.3 Robot

The robot which carries the sensors, as seen in figure 3.1, is a Mitsubishi PA10, an industrial, position controlled 7-Degrees-of-Freedom (DOF) robotic arm.

3.2 Data Collection

During the collection of the training data we synchronously record the data provided by the BioTac and the force-torque sensor, while applying force to the BioTac. The presence of the force-torque sensor is justified with the need to generate training output data and to have a ground truth for comparison while testing.

During the recording of the data we manually pushed with a solid wooden obstacle against the BioTac sensor as shown in figure 3.4. The goal was to keep the variation of the contacts position and angle over the trials as small as possible. We are aware of the fact that we can not eliminate these kind of variations completely but we think, that this way of data collection has a higher correlation to real world applications and therefore has a positive influence of the interpretation of our results in chapter 4.

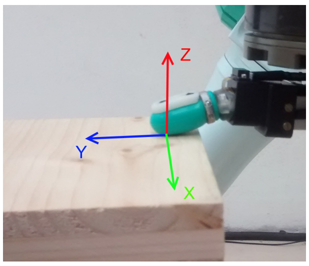


Figure 3.4: Contact situation during data collection with added frame

We collected our data at a rate of 100 Hz, which is the BioTacs maximum sampling rate, and should be high enough for this kind of quasi-static application. To ensure this we performed all movements during the data collection at slow speeds.

To collect training data which enables us to test our algorithm we created four data sets. In the first three data sets we tried to apply mainly forces in one direction. The maximum forces and torques per set and dimension are given in

	F_x [N]	F_y [N]	F_z [N]	T_x [Nm]	T_y [Nm]	T_z [Nm]
Set 1	5.20	1.83	5.52	2.29	0.43	1.19
Set 2	1.72	6.28	5.74	1.16	0.33	0.94
Set 3	0.96	1.34	10.59	2.13	0.09	0.28
Set 4	6.53	7.5	10.84	1.14	0.06	0.19

Table 3.1: Maximum force per dimension occurring in each of the training sets

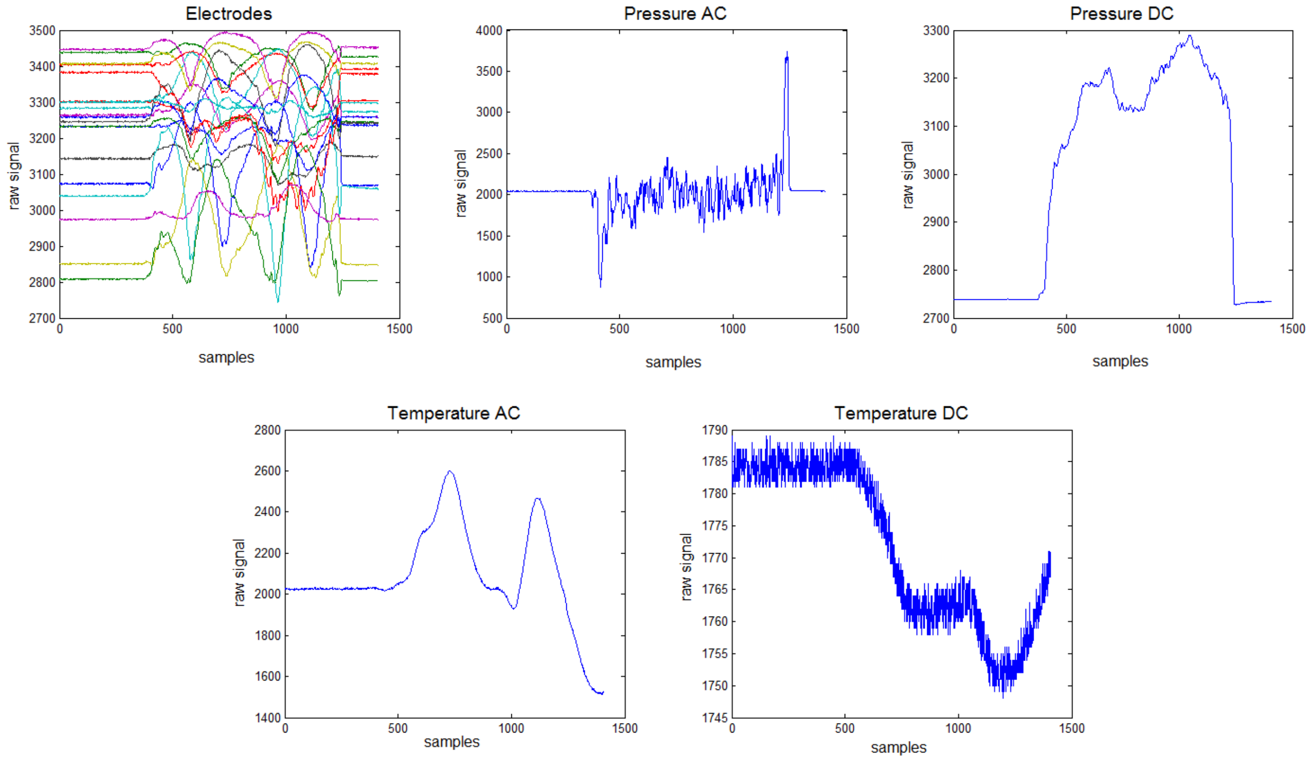


Figure 3.5: Sample output data of the 4th set provided by the BioTac

table 3.1. The values for set 1 and set 2 indicate that we were forced to apply a relative large amount of force in z direction while pushing in x or y direction to avoid slip between the BioTac and the wooden plate.

We performed 10 trials per set, each lasting about 20 seconds. One trial consisted of 2 two pushes in the corresponding direction for the single force sets. For the combined set we rotated the applied force in the x-y plane and varied the contact force in z direction. A sample of the resulting data of a combined set is shown in figure 3.5 for the BioTac signals and in figure 3.6 for the corresponding force-torque signals.

The first subplot in the upper row of figure 3.5 shows the different feedback each electrode provides under the applied force. The second subplot shows the vibrations of the fluid during the trial and the third subplot shows the overall pressure in the fluid. The acquired temperature information is plotted in the lower row of figure 3.5.

Correlated to the data plotted in figure 3.5, figure 3.6 shows the forces recorded by the force-torque sensor. The first row of plots shows forces with respect to (w.r.t.) the dimension. The first two subplots illustrate the rotation of the applied force in the x-y plane. The second row shows the corresponding torques, which are relatively small compared to the forces.

3.3 Comparison Method: Neural Network

We have implemented the Neural Network (NN) described in [14] with the use of MATLAB's Neural Network Toolbox. As illustrated in figure 3.7 the NN consists of three layers. The size of the input layer is determined by the number of electrodes as 19. The hidden layer size is set to 38, which is the recommended in [14]. Being two times the stated size of number of inputs. The Neural Network should give us a prediction for the occurring forces or torques in all directions.

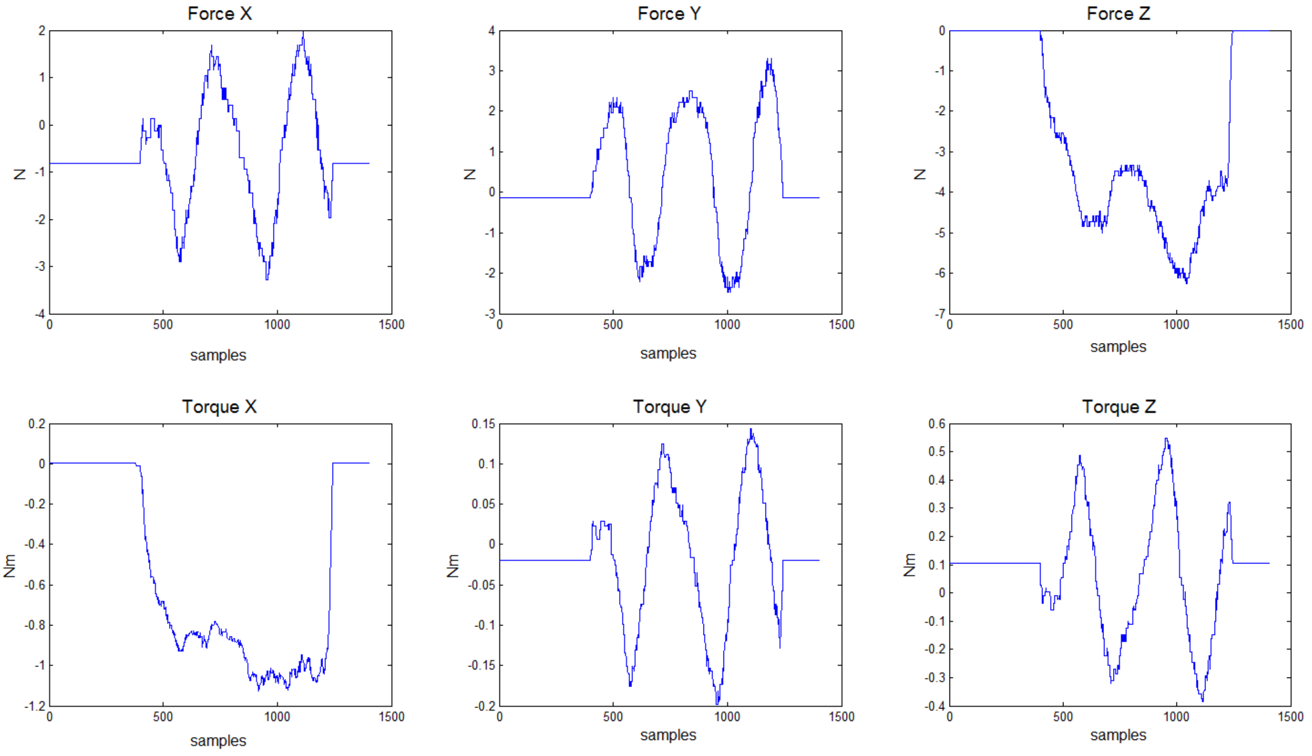


Figure 3.6: Sample output data of the 4th set provided by the force-torque sensor

Therefore the size of the output layer is set to 3. As described by Wettels and Loeb the NN uses the Levenberg-Marquardt back propagation algorithm to fit the model predictions to the data. Hyperbolic tangent activation functions are used by the hidden units and linear functions are used by the outer units.

The described overfitting prevention through early stopping and Bayesian regularization as described by Wettels and Leob is also employed in our NN setup.

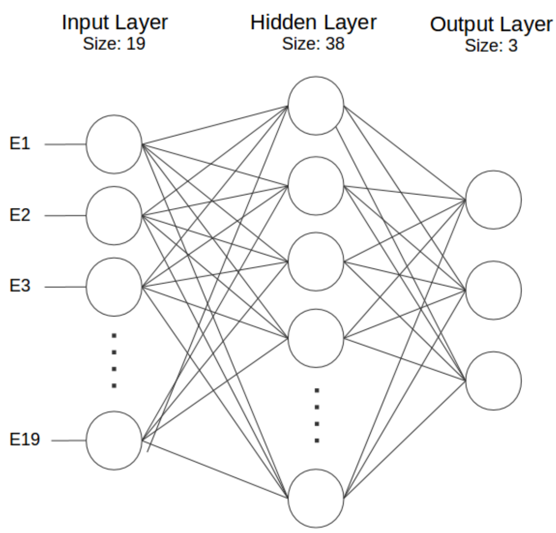


Figure 3.7: Visualisation of the implemented Neural Network with given size of the layers

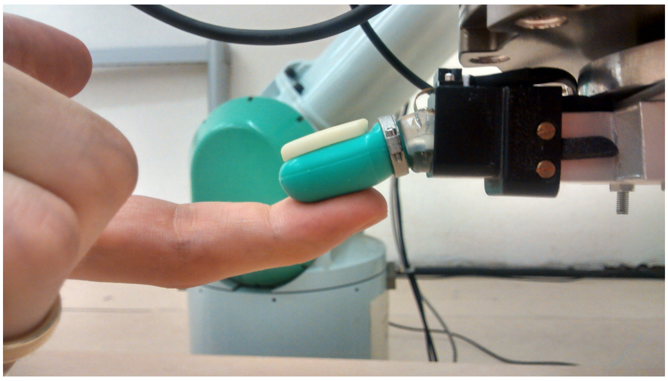


Figure 3.8: Contact situation between the participant and the BioTac during the experiment

3.4 Training and Validation

We perform 2 evaluations. First we compare a prediction based on the pressure signal to a model trained on the electrode signals. The second evaluation compares the GP with the NN.

Pressure Signal vs Electrode Signals

To enable us to judge if we benefit from the usage of the data provided by the electrodes we compared the predictions of a GP trained on electrode data with one trained on the pressure signal provided by the BioTac.

Due to the fact that the pressure is just representing the overall pressure applied to the liquid it does not give any information about the direction the force is applied. We try to take care of this constraint by using the data set for training where we are mostly producing force in one direction and reduce the prediction output to this dimension. According to table 3.1 we used data set 3 for training and testing the two models.

We apply 5-fold cross validation on the chosen data. This means we are using 8 trials from data set 3 as training data and the remaining 2 trials as testing data. This is done 5 times so that every trial was once included in the testing data.

Gaussian Process vs Neural Network

To compare the performance of the Gaussian Process with the Neural Network we train and test a GP and a NN on exactly the same data.

Because the arising torques are very small compared to the forces and the hyper-parameters of the GP can not be fitted to both without a loss of performance we train separate models for them. For this experiment we do not need to reduce the training output dimensionality because both methods predicting forces or torques in all three directions.

We apply the described validation method over all of our recorded data. This means the training data is a concatenation of eight trials out of each set and the test data contains the remaining two trials out of each set.As a measure of performance and for comparison we use the root-mean-squared error (RMSE) which is computed in each iteration of the cross validation.

3.5 Demonstration

We have performed 2 different tasks to demonstrate that our method can be applied in a real scenario. The task in the first demonstration for the robot/control is to move until contact with an object occurs and then maintain contact at a desired force. We reduced the allowed movements and thus also the predictions for this task to the z direction. To set up the first task we placed an object under the end-effector (BioTac) of the robot. The robot should now move downwards and stop when he touches the object to avoid damaging it.

We used a given PD controller which takes the predicted forces from our GP and the desired forces. For this application we set the desired force in z direction to 1 N. This value is small enough to avoid damaging all objects we used for testing but. We performed several trials with different objects each with different degree of compliance like a wooden box, a carton box and a stuffed animal.

The second task for the robot/control is to maintain contact with the participants fingertip while the participant is moving his finger in space. The movements performed by the human are limited to translations, that means the finger needs to stay in the same orientation while moving.

Figure 3.8 shows the contact situation between the robot and the participant during the experiment. For example when the human is pushing upwards against the BioTac and exceeds the desired force the robot should move upwards to compensate the increased force. The same behaviour is applied to the other dimensions.

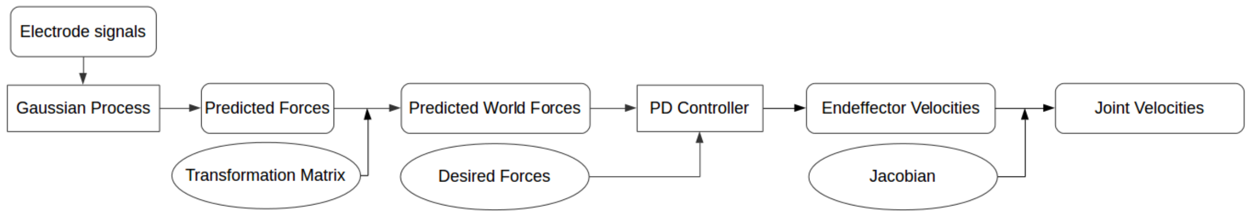


Figure 3.9: Diagram of the information flow

When the human pushes in a horizontal direction the robot should follow with the appropriate movement. As seen on figure 3.8 the contact is very different to the contact situation while collecting data shown in figure 3.4. Therefore we collected new set of application related training data where we used our finger to stimulate to sensor. We applied forces related to the described forces applied in the validation data sets in each of the 4 trials of the new set.

The contact forces the robot should maintain are given on initialisation and consist of one value for each dimension. The desired forces in x and y directions are set to zero to make sure the robot is only moving in this plain when the participant is applying forces in this plain. We have chosen $F_z = 1$ N which means, that the robot pushes in vertical direction onto the finger and therefore maintains gentle contact even when the participant is moving downwards.

The employed PD controller adapts the robots movements accordingly when the difference between the desired and applied force reach a threshold of 0.2 N. The PD controller computes the resulting endeffector velocities with respect to a gain factor. The endeffector velocities are transformed via a Jacobian to joint velocities. Figure 3.9 illustrates the described information flow during the demonstrations.

4 Results and Discussion

We have generated several results that will be shown in this chapter. First we will present the results of the comparison of the pressure signal trained Gaussian Process (GP) with a GP trained on the electrodes feedback. After that we will state the results of the comparison of the GP with the Neural Network (NN) and we will close with the results of the performed demonstrations.

Pressure Signal vs Electrode Signals

The 5-fold cross validation on the force prediction in z direction based on the pressure input yield to a root-mean-squared error (RMSE) of 0.92 N which is not precise when considering the maximum applied force in z direction of 10.8 N. This is an relative error of 8.5% and also the resulting RMSE of the prediction based on electrode input of 0.35 N confirms the bad performance. In figure 4.1 the predictions based on the pressure signal and on the electrode signals are plotted. The figure shows the large variance of the pressure signal based prediction compared to the variance of the electrode signal based prediction.

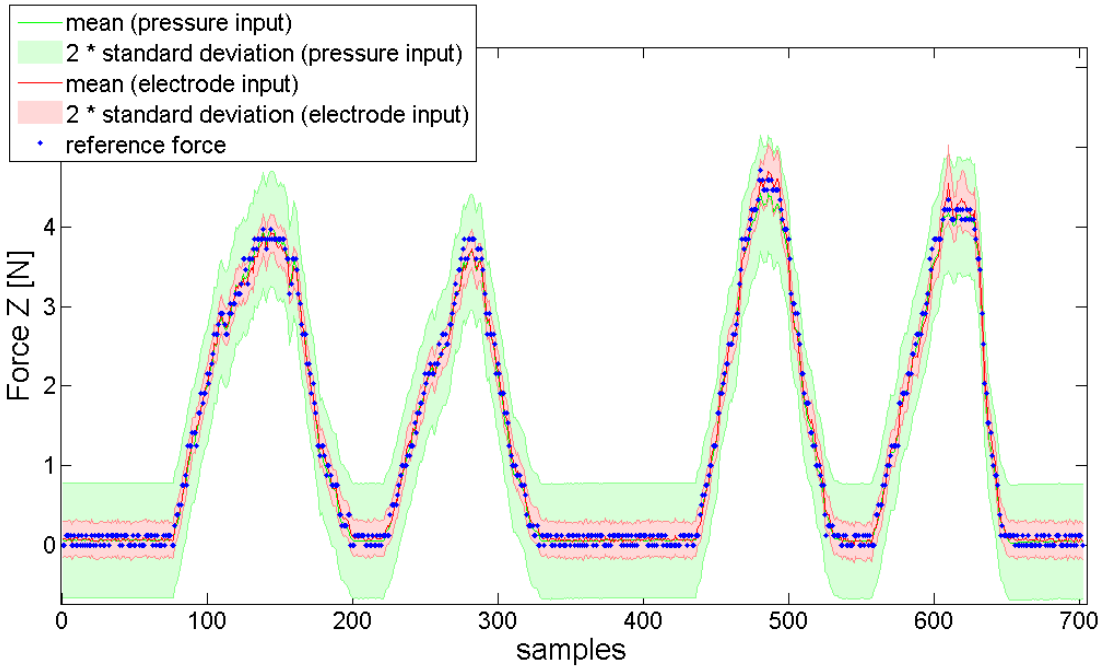


Figure 4.1: Comparison of a model trained on pressure signal with a model trained on electrode signals

We assume, that the poor performance is caused by our way of data collection in combination with the fact, that the pressure signal can not differ between the location or direction the applied force.

To be more precise, during data collection we unintentionally have applied the forces on slightly different locations on the sensor and pushed from a marginal different angle, which results in different deformations of the skin. These different deformation will produce different pressures and therefore the mapping from pressure to force is not that precise.

Furthermore as shown in figure 4.2 the correlation between the pressure signal and the applied force is decreasing drastically when increasing the force. This behaviour can be the consequence of complete saturation of the pushes. This means the distortion of the fluid is only dependent on the angle and the position the obstacle is pushed against the sensor and no longer on the amount of force. This phenomena also contributes to the poor performance of the prediction.

Nevertheless the pressure signal is reflecting small changes in the overall applied force. Therefore it can be a good indicator for the presence of contact and can for example be used in an contact detection application.

Gaussian Process vs Neural Network

The generalized cross validation results of the Gaussian Process force prediction are presented in table 4.1. Considering the range of the arising forces which are indicated by table 3.1 the results seems reasonable. For the prediction

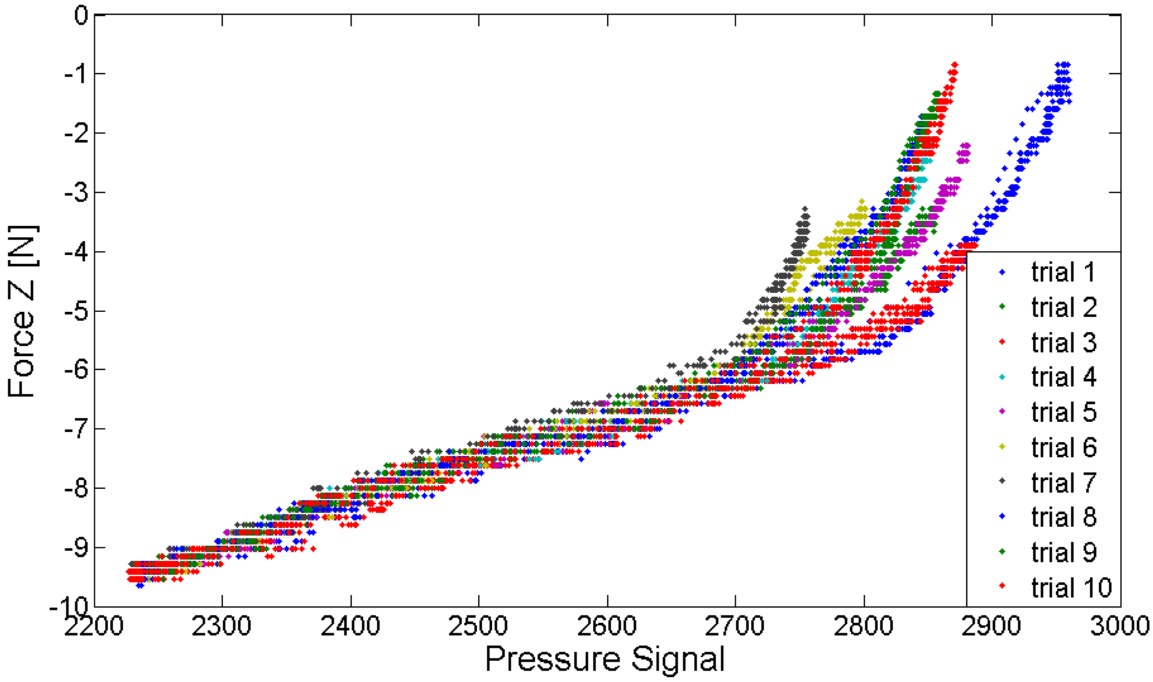


Figure 4.2: Pressure signal plotted over force signal.

	F_x [N]	F_y [N]	F_z [N]
Gaussian Process	0.266	0.279	0.232
Neural Network	0.277	0.283	0.229

Table 4.1: Results of the force prediction validation, generalized over the 5 validation sets (RMSE)

in z direction a RMSE of 0.24 N which is equal to 24 g when thinking of weights. When considering the described demonstrations in chapter 3.5 the results seem suitable to perform this tasks.

The predictions of the Gaussian Process during one of the validation iterations are plotted in figure 4.3. The figure allows us to verify the validation results visually. The prediction of the GP plotted in green stays always very close to the reference force plotted in black over the whole test set.

Compared to results obtained from the validation of the Neural Network we can not state any significant performance advantages of one model. The results of the generalized 5-fold cross validation presented in table 4.1 for the NN and the GP are very similar. Also the validation results of each iteration given in table 4.2 indicate a equal performance of the methods. Additionally in figure 4.3 it is hard determine any notable difference of the predictions. The only notable behavioural variations occur at the load peeks where the NN tends to slightly predict higher values.

When interpreting the resulting values, given in table 4.3, of the cross validation applied on the GP model torque prediction it is important to consider table 3.1. The maximal arising torques are very small and therefore the RMSE 0.39 Nm is huge.

In figure 4.4 the torque predictions and the reference torques are plotted for each dimension in separate plots. The figure illustrates the poor performance of the prediction model. The predictions of the GP plotted in green are in some regions relatively far of the real values, especially in regions where torques occur and predictions remain zero. This is a comprehensible behaviour considering the composition of the training data with very rare samples actually representing the occurrence of torques.

We emphasize that the data was created with the goal of producing forces and that we did not create training data explicitly for torques. Also our data collection setup as shown in figure 3.4 is not suitable to apply torques onto the sensor. Nevertheless we infer from figure 4.4 that the torque prediction of our GP model is capable of providing information about whether torques occur and also indicate the orientation and direction of those torques..

The validation results of the NN model predicting torques indicate a significant worse performance compared to the GPs results. In figure 4.4 the corresponding predictions of the NN are presented. When no torques occur the NN

		F_x [N]	F_y [N]	F_z [N]	T_x [Nm]	T_y [Nm]	T_z [Nm]
Iteration 1	Gaussian Process	0.241	0.319	0.201	0.399	0.049	0.402
	Neural Network	0.260	0.279	0.205	0.393	0.016	0.387
Iteration 2	Gaussian Process	0.289	0.246	0.238	0.403	0.040	0.359
	Neural Network	0.342	0.281	0.221	0.506	0.063	0.500
Iteration 3	Gaussian Process	0.296	0.237	0.167	0.405	0.059	0.434
	Neural Network	0.280	0.235	0.172	0.440	0.014	0.445
Iteration 4	Gaussian Process	0.246	0.315	0.248	0.357	0.057	0.424
	Neural Network	0.248	0.308	0.248	0.448	0.017	0.464
Iteration 5	Gaussian Process	0.258	0.280	0.304	0.380	0.017	0.379
	Neural Network	0.256	0.312	0.298	0.394	0.020	0.389

Table 4.2: Detailed cross validation results (RMSE)

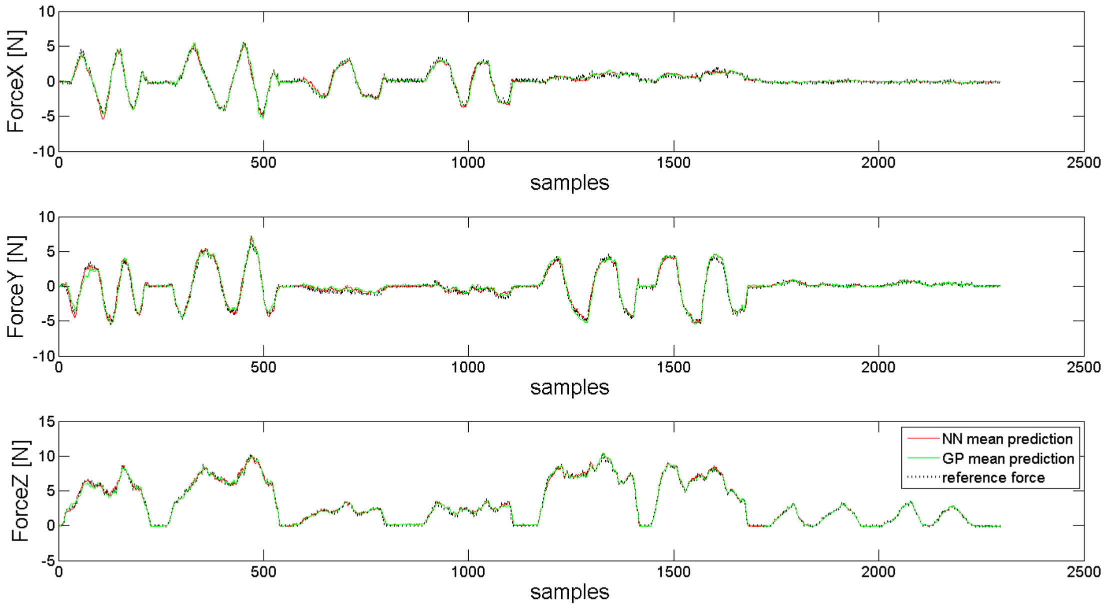


Figure 4.3: Force predictions of both models compared to reference torques.

	T_x [Nm]	T_y [Nm]	T_z [Nm]
Gaussian Process	0.389	0.044	0.399
Neural Network	0.437	0.026	0.437

Table 4.3: Results of the torque prediction validation, generalized over the 5 validation sets (RMSE)

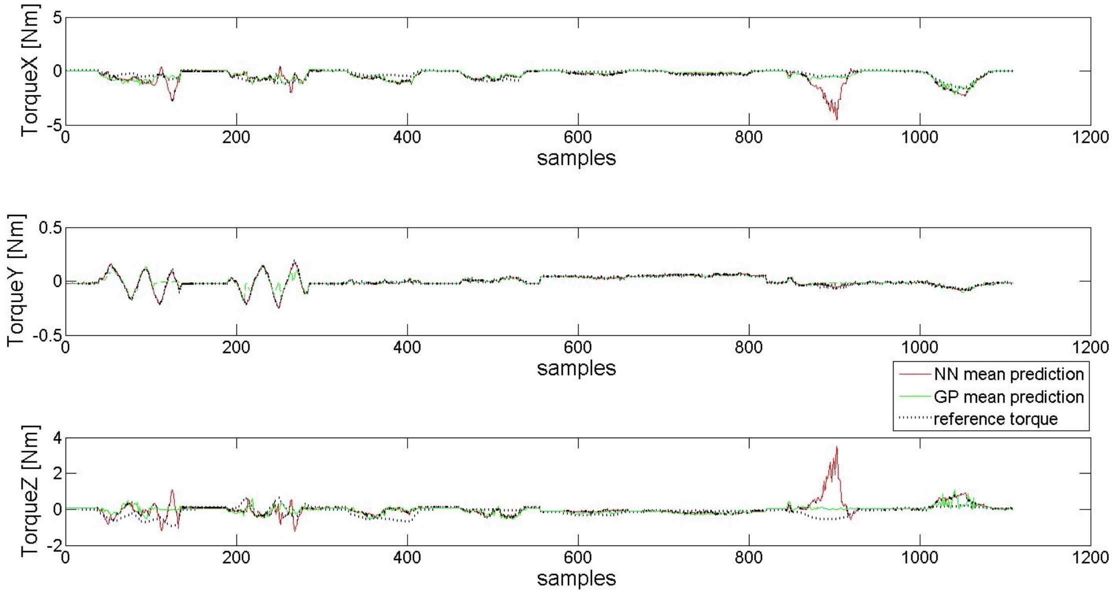


Figure 4.4: Torque predictions of both models compared to reference torques

predictions are as good as the GP's predictions. Differently to the behaviour of the GP, the NN tend to overshoot when torques arise. In the given prediction plots for torques around the x and z axis the NN drastically over estimates the arising torques.

But as the data collection procedure suggests, the focus of this thesis is more on force prediction. Therefore we can not infer any results regarding the methods performance and recommend further study on torque extraction from feedback provided by the BioTac sensor.

Demonstrations

The recorded force data obtained by the force-torque sensor and the predictions recorded during the guarded movement demonstration are given in figure 4.5. The force predictions of our model are plotted in red and the measured forces from the force-torque sensor are given in blue. In the first few seconds the robot moves straight downwards until it reaches the obstacle. In this section the noise of the signal can be observed. The signals provided by the force torque sensor react to vibrations of the system and show high noise values. The predicted forces based on the BioTac feedback show very small signal variations. This behaviour remains over the course of the whole experiment and the noise of the measured force signal grows when contact is made. We assume the small movements the robot performs while maintaining the force result in even higher system vibrations and therefore in more noise.

When the robot reaches the obstacle both the predicted and the measured forces are responding with a rising value. As shown in figure 4.5 the force on initial contact exceeds the desired value of 1 N but this is a control related problem and not the focus of this thesis. To reduce the overshoot at initial contact a slower robot velocity could be a solution.

Until the end of the demonstration, the predicted and the measured forces vary around the desired force which is the expected behaviour.

The recorded data for the joystick demonstration is given in figure 4.6. The predicted and measured forces are plotted for each dimension separately and for each direction the desired force is indicated by a horizontal dashed line. The plots show that the predictions, in most regions, are close to the forces measured by the force-torque sensor. This reflects the subjective impressions during the demonstration. The robot followed the movements very gently in all directions and the maintained force was appropriate throughout the demonstration.

Solely movements in negative y direction, corresponding to figure 3.4, did not perform well. The appliance of force in this direction resulted in an robot movement in the negative y direction but additionally the robot did also moved in z direction. We assume that this behaviour arises from the orientation the BioTac is mounted on the force-torque sensor as shown on figure 3.8. The tip of the sensor pointing downwards which makes it difficult to apply forces in the negative y direction without generating forces in z direction.

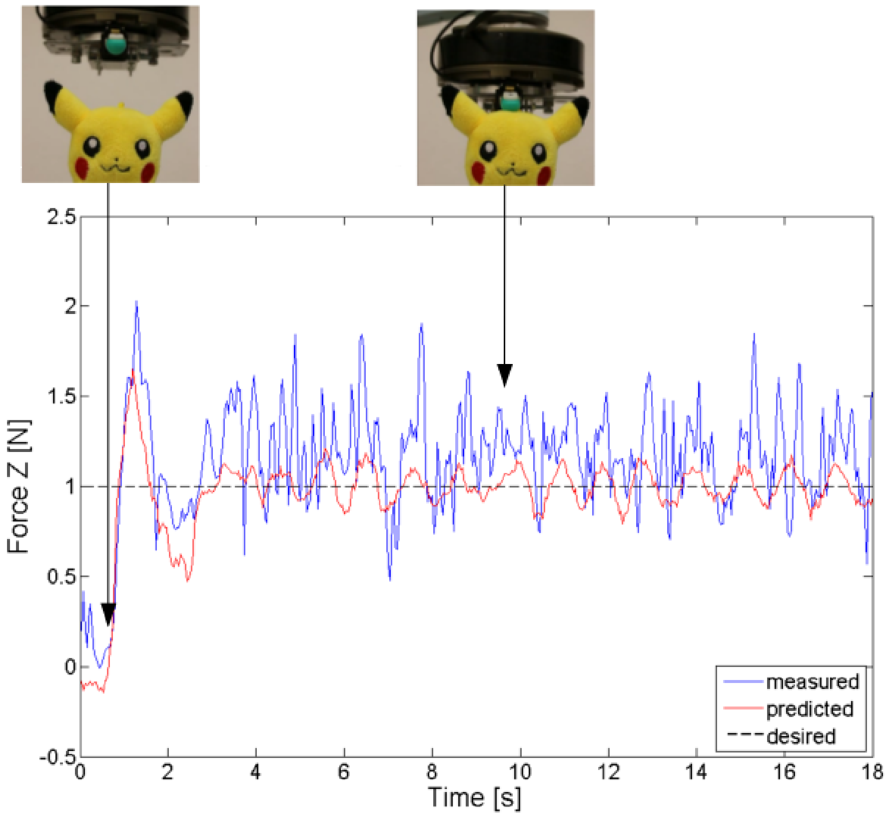


Figure 4.5: Predicted, measured and desired forces arising during the guarded movement demonstration with images

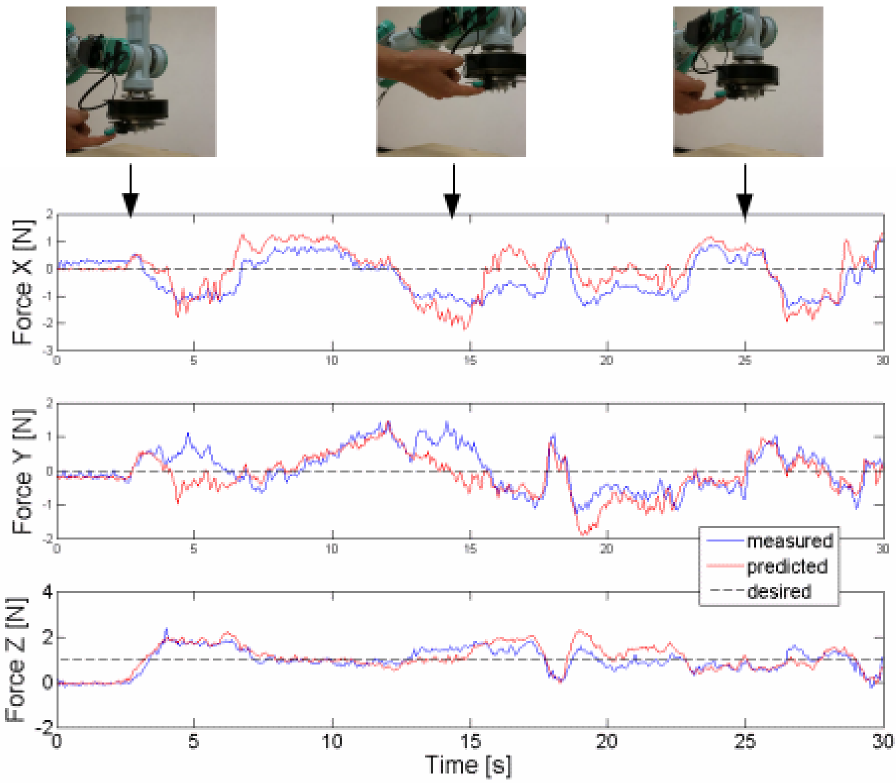


Figure 4.6: Predicted, measured and desired forces arising during the joystick demonstration with images

5 Conclusion and Future Work

The motivation behind this thesis was to make a contribution to the field of object manipulation and therefore study contact based feature extraction from data provided by the BioTac, a human inspired fingertip sensor. In this thesis we have successfully extracted forces and torques from tactile feedback provided from the BioTac sensor using a Gaussian Process (GP) regression. We validated our method and compared it's performance to a Neural Network (NN). Furthermore we performed two demonstrations which indicate possible applications of our method.

We have implemented a GP which is able to process multiple input and output dimensions and applies hyper-parameter optimisation by maximising the marginal likelihood.

We started with a comparison between predictions based on the pressure signal and predictions based on signals provided by the electrodes. This comparison was limited to forces in z direction due to the higher correlation in this dimension. The results indicated that the usage of the pressure signal is a capable of providing a reasonable force prediction but gets outperformed by models which make use of the electrode signals. We suggested the use of the pressure signal rather as an indicator for contact than for precise force determination.

From then on we focused on the use of data provided by the BioTac's nineteen electrodes. We have trained separate models for force and torque prediction due to the very different characteristics of the signals and the incapability to optimise the hyper-parameters to both signals at once.

We tested and trained our models on the collected data and used the root-mean-squared error (RMSE) as a measure of performance. To be able to further judge the performance of the applied GP we implemented the already mentioned NN stated in [14] as comparison method. The results of the applied 5-fold cross validation indicated that both methods perform roughly equally well. The force predictions were accurate but the predictions of the torques were not performing well. We need to recognize that when collecting data we did not specifically aim for torque data and recommend further studies here.

We have transferred our method to the robot and successfully performed a 'stopping' demonstration where task was to move downwards until contact with an obstacle is made and then maintain a given desired force. The results indicated that our prediction model is capable of providing reliable force prediction and therefore prevent deforming or crushing objects.

Based on the guarded movement task in the final joystick demonstration the robot should follow a participants finger movement while maintaining a gentle contact force. The measured forces during the execution proved, that the prediction and the controller are working correctly. The subjective impressions, throughout the demonstration where throughout positive, despite the fact that guiding the robot backwards did not work perfectly. We assumed that this misbehaviour is caused by the setup which did not allow to guide the robot backwards without the appliance of force upwards, causing the robot to also move upwards.

With the performed demonstrations we showed the applicability of our method and that we are able to reliable determine contact forces. They also indicate the broad possible application area. The stopping application can be adapted to work in other directions and thus enable a robot to for example pinch an object against a wall. With the joystick demonstration we showed that it is possible to implement a active compliance relying only on the BioTac signals, this can be useful in applications where guiding a robot over a trajectory or in general in scenarios where humans and robots act close together.

Furthermore the overall performance of our GP model could maybe be improved by for example employing a different kernel function. We did only apply a Gaussian kernel in our model, but it could be that there are other kernel functions that better fit to the data provided by the BioTac.

The precision of our force determination method needs to be improved further to be able to be keep up with the human sensitivity of pressure differences of applied forces [18].

Bibliography

[1] A. Faisal, D. Stout, J. Apel, and B. Bradley, "The manipulative complexity of lower paleolithic stone toolmaking," *PloS ONE*, vol. 5, no. 11, p. e13718, 2010.

[2] M. Grebenstein, M. Chalon, W. Friedl, S. Haddadin, T. Wimböck, G. Hirzinger, and R. Siegwart, "The hand of the DLR hand arm system: Designed for interaction," *The International Journal of Robotics Research*, vol. 31, no. 13, pp. 1531–1555, 2012.

[3] Shadow Robot Company, "The shadow dextrous hand.," tech. rep., 2014.

[4] C. Lovchik and M. A. Diftler, "The robonaut hand: A dexterous robot hand for space," in *Robotics and Automation*, vol. 2, pp. 907–912, IEEE, 1999.

[5] S. Schulz, C. Pylatiuk, and G. Bretthauer, "A new ultralight anthropomorphic hand," in *Robotics and Automation*, vol. 3, pp. 2437–2441, IEEE, 2001.

[6] *BioTac Product Manual*, v16 ed., February 2013.

[7] S. Chitta, J. Sturm, M. Piccoli, and W. Burgard, "Tactile sensing for mobile manipulation," *Transactions on Robotics*, vol. 27, no. 3, pp. 558–568, 2011.

[8] J. M. Romano, K. Hsiao, G. Niemeyer, S. Chitta, and K. J. Kuchenbecker, "Human-inspired robotic grasp control with tactile sensing," *Robotics, IEEE Transactions on*, vol. 27, no. 6, pp. 1067–1079, 2011.

[9] Z. Su, J. A. Fishel, T. Yamamoto, and G. E. Loeb, "Use of tactile feedback to control exploratory movements to characterize object compliance," *Frontiers in neurorobotics*, 2012.

[10] V. Chu, I. McMahon, L. Riano, C. G. McDonald, Q. He, J. M. Perez-Tejada, M. Arrigo, N. Fitter, J. C. Nappo, T. Darrell, *et al.*, "Using robotic exploratory procedures to learn the meaning of haptic adjectives," in *Robotics and Automation (ICRA)*, pp. 3048–3055, IEEE, 2013.

[11] A. L. Trejos, J. Jayender, M. Perri, M. D. Naish, R. V. Patel, and R. Malthaner, "Robot-assisted tactile sensing for minimally invasive tumor localization," *The International Journal of Robotics Research*, 2009.

[12] T. Van Anh Ho, Nagatani, A. Noda, S. Hirai, *et al.*, "What can be inferred from a tactile arrayed sensor in autonomous in-hand manipulation?," in *Automation Science and Engineering (CASE)*, pp. 461–468, IEEE, 2012.

[13] N. Wettels, J. Fishel, Z. Su, C. Lin, and G. Loeb, "Multi-modal synergistic tactile sensing," in *Tactile sensing in humanoids—Tactile sensors and beyond workshop, 9th IEEE-RAS international conference on humanoid robots*, 2009.

[14] N. Wettels and G. Loeb, "Haptic feature extraction from a biomimetic tactile sensor: Force, contact location and curvature," in *Robotics and Biomimetics (ROBIO)*, pp. 2471–2478, IEEE-RAS, Dec 2011.

[15] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. MIT Press, 2006.

[16] C. M. Bishop, *Pattern recognition and machine learning*. Springer New York, 2006.

[17] *Force-torque Sensor Type FTC / FTCL, Assembly and Operating Manual*, November 2008.

[18] W. Morley, *Neural Aspects of Tactile Sensation*. North-Holland, 1998.

[19] D. Barber, *Bayesian Reasoning and Machine Learning*. Cambridge University Press, 2012.

[20] SynTouch LLC, *BioTac - The Future of Machine Touch*, 2013.

[21] M. Jägersand, O. Fuentes, and R. Nelson, "Acquiring visual-motor models for precision manipulation with robot hands," in *Computer Vision—ECCV'96*, pp. 603–612, Springer, 1996.