Technische Universität München
Fakultät für Elektrotechnik und Informationstechnik
Lehrstuhl für Informationstechnische Regelung

Human Motor Behavior Prediction through Gaussian Process Modeling on Manifolds

Muriel Friederike Lang

Vollständiger Abdruck der von der Fakultät für Elektrotechnik und Informationstechnik der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktor-Ingenieurs (Dr.-Ing.)

genehmigten Dissertation.

Vorsitzender: Prof. Dr.-Ing. Klaus Diepold

Prüfende der Dissertation:

1. Prof. Dr.-Ing. Sandra Hirche
2. Prof. Dongheui Lee, Ph.D.

Die Dissertation wurde am 18.09.2018 bei der Technischen Universität München eingereicht und durch die Fakultät für Elektrotechnik und Informationstechnik am 12.03.2019 angenommen.

Preamble

This thesis summarizes research results from six years of work; first, at the Chair of Automatic Control Engineering (LSR) and later, at the newly founded Chair of Information-Oriented Control (ITR) at the Technical University of Munich.

During my time at the institute, I had the pleasure to meet and work with a number of brilliant researchers who inspired and supported me. I would like to express my gratitude to those people for making my time as PhD remarkable. First and foremost, I would like to thank my doctoral advisor and head of the Chair of Information-Oriented Control, Prof. Dr. Sandra Hirche, for her support in accomplishing the necessary scientific work that led to the present results. Without her dedication to excellence in research and the rich and multidisciplinary scientific exchange initiated and coordinated by her, this thesis would never have been possible. My work profited deeply from the international network of collaborators of our institute and the thereof resulting fruitful research discussions.

Likewise, I would like to thank all my colleagues and teammates for numerous scientific exchange and for their prompt help at any time. Those people, however, not only helped in scientific questions, but made life at ITR an enjoyable and diverting one. Many thanks as well to the administration and technical staff for providing support whenever needed. Finally, I would like to thank my family and friends for encouraging me to continue my scientific career and for their enduring support in all concerns during my PhD.

Munich, September 2018 Muriel Lang

Acknowledgments

The research leading to these results has received funding from the EU Horizon2020 project RAMCIP, under grant agreement n° 643433, the ERC Starting Grant “Control based on Human Models (con-humo)" under grant agreement n° 337654 and the European Union Seventh Framework Programme FP7/ 2007-2013 under grant agreement n° 601165 of the project “WEARHAP - WEARable HAPtics for humans and robots".

Abstract

Understanding and anticipating human motor action in unconstrained environments is a challenging task, even for other humans, if there is no precise task description. It is essential, however, in many application areas, such as healthcare and rehabilitation for instance. Further, the prediction of motor behavior, is a pre-requisite for successful and intuitive human-robot interaction. Therefore, we present in this thesis a holistic plant model for the complex human sensorimotor system to improve the prediction quality of human motor behavior. Our dynamics model includes all relevant aspects, specifically the internal decision making, the motor execution and the influence of motor dysfunctions. To consider each of the aspects in the novel dynamics model, we employ a data-driven approach, as this is capable of representing system behavior to which first-order principles cannot be employed. For approximating human motor behavior based on observations the Gaussian process model is especially suitable, as this technique accounts for uncertainty in the model.

Part of the sensorimotor system dynamics evolve in the space of rigid motions, which constitute the space of natural human behavior. Unfortunately, this space is nonlinear. So far, the available learning approaches cannot handle dynamics evolving in the space of rigid motions. Dynamics evolving in a nonlinear space, such as rotations for instance, are not appropriately considered in well-known data-driven approaches. Therefore, we present in this thesis a mathematical framework for Gaussian processes, where the valid input domain is generalized to full rigid motions, namely the special Euclidean group $SE(3)$. This generalized Gaussian process model is suitable for approximating human motor execution. The correctness and accuracy of our approach is validated on simulated and real human motion data. We analyze the estimation performance of the novel Gaussian process framework by comparing to state of the art techniques, and show significantly improved prediction accuracy for rigid motions.

Further, we employ the findings of Gaussian process modeling on $SE(3)$ in probabilistic trajectory classification on $SE(3)$ and provide an online algorithm for human movement prediction based on the generalized Gaussian process. In the process of generating the motion, the classification results are taken into account for directing the prediction into the most probable direction. This approach approximates the decision making among multiple potential movement plans in the human sensorimotor system. The algorithm outperforms the human ability to predict other human's motion intention in speed and accuracy.

Finally, the motor state of the human system is estimated. This approach is used to detect faults in the sensorimotor system, such as motor dysfunctions caused by neurodegeneration. We propose a method for monitoring Parkinson's disease symptoms using a wearable motion tracking device which can recognize and predict symptoms during daily living. In contrast to alternative approaches, we do not restrict the patients' activities during symptom monitoring as our approach is capable of differentiating voluntary from unintended motions with high accuracy. Experimental validation of our approach on real patient data demonstrates a significant accordance of the estimations with medical expert assessment.

Together, the presented techniques allow for holistic consideration of the human motor behavior, as they jointly approximate our plant model of the human sensorimotor system.

Zusammenfassung

Das Verstehen und die Vorhersage menschlicher Bewegungen in unbeschränkter Umgebung ist selbst für andere Menschen schwierig, wenn die Aufgabe nicht im Detail bekannt ist. In vielen Anwendungsgebieten, wie beispielsweise im Gesundheitswesen und der Rehabilitation ist es jedoch essentiell. Auch die Vorhersage von motorischem Verhalten ist eine Voraussetzung für erfolgreiche und intuitive Mensch-Roboter Interaktion. Daher stellen wir in dieser Dissertation ein holistisches Modell für das komplexe menschliche sensomotorische System vor, um die Vorhersagegenauigkeit des Bewegungsverhaltens zu verbessern. Unser dynamisches System beinhaltet alle relevanten Aspekte, von der internen Entscheidungsfindung über Bewegungsausführung bis hin zum Einfluss von motorischen Störungen. Um jeden dieser Aspekte im neuen dynamischen Modell zu berücksichtigen, verwenden wir einen datengetriebenen Ansatz, da er Systemverhalten darstellen kann, für das Lösungsansätze erster Ordnung nicht anwendbar sind. Um menschliches Verhalten basierend auf Beobachtungen zu approximieren, ist besonders der Gaußprozess geeignet, da dieses Verfahren Unsicherheit im Modell berücksichtigt.

Menschliche Bewegungsdynamik beinhaltet Starrkörperbewegungen, die den Raum der natürlichen menschlichen Bewegungen definieren. Ungünstigerweise ist dieser Raum nichtlinear. Bisher können die vorhandenen Lernverfahren Bewegungen in diesem Raum nicht handhaben. Also werden Rotationen in den bekannten datengetriebenen Ansätzen nicht angemessen berücksichtigt. Daher stellen wir in dieser Dissertation ein mathematisches Framework für Gaußprozesse vor, in dem der gültige Definitionsbereich auf die spezielle euklidische Gruppe $SE(3)$ erweitert wird. Die Richtig- und Genauigkeit unseres Ansatzes wird auf simulierten und echten Menschendaten überprüft. Wir analysieren die Schätzungsperformanz des neuen Frameworks im Vergleich zum Stand der Technik und zeigen dabei eine signifikant verbesserte Vorhersage von Starrkörperbewegungen auf.

Außerdem wenden wir die Ergebnisse über Gaußprozessmodellierung in $SE(3)$ auf probabilistische Trajektorienklassifierung in $SE(3)$ an und entwickeln einen Algorithmus zur online Vorhersage menschlicher Bewegung. Bei der Bewegungsgenerierung werden die Ergebnisse der Klassifizierung berücksichtigt. Dieser Ansatz nähert die Entscheidungsfindung zwischen verschiedener potentieller Bewegungspläne im sonsomotorischen System an. Der entwickelte Algorithmus übertrifft die menschlichen Fähigkeiten, die Bewegungsintention anderer Menschen vorherzusagen.

Schließlich wird der Zustand des motorischen Systems geschätzt, um Fehler im sensomotorischen System zu erkennen, wie etwa durch Neurodegeneration verursachte motorische Störungen. Wir präsentieren einen Ansatz um Parkinson Symptome im alltäglichen Leben zu erkennen und vorherzusagen. Wir verwenden einen tragbaren Bewegungsmesser zur Aufzeichnung der Symptome. Im Gegensatz zu alternativen Ansätzen beschränken wir die Aktivitäten der Patienten währen der Symptomüberwachung nicht, da unser Ansatz gewollte und unbeabsichtigte Bewegungen präzise unterscheiden kann. Die experimentelle Auswertung unseres Ansatzes auf echten Patientendaten zeigt eine signifikante Übereinstimmung unseres geschätzten Zustands mit professioneller medizinischer Bewertung.

Durch diese Techniken lässt sich unser Modell des sensomotorischen Systems ganzheitlich annähern.

Contents

Notation **ix**

1 Introduction **1**
1.1 Challenges . . . 2
1.2 Human Motor Behavior . . . 3
1.3 Data-Driven Modeling . . . 4
1.4 Main Contribution and Outline of the Thesis . . . 5

2 Dynamics Model of the Sensorimotor System **9**
2.1 Human Motion Dynamics on $SE(3)$ . . . 13
2.2 Decision Process Dynamics . . . 14
2.3 Motor State Dynamics . . . 16
2.4 Discussion . . . 18

3 Gaussian Process for Rigid Motion Dynamics on $SE(3)$ **21**
3.1 Related Work and Open Problems . . . 21
3.2 Gaussian Process Definition and Characteristics . . . 23
  3.2.1 Model training . . . 24
  3.2.2 Prediction . . . 25
  3.2.3 Classic 6-DoF Gaussian Process . . . 25
3.3 Representations of $SE(3)$ . . . 25
  3.3.1 Dual Quaternions . . . 27
  3.3.2 Axis-angle and Translation Vectors . . . 29
3.4 Gaussian Process Generalization . . . 30
  3.4.1 Mathematical Basis Concept . . . 30
  3.4.2 Metrics and Distance Measures . . . 31
  3.4.3 Mean and Kernel Functions . . . 34
3.5 Integration from Velocity Space to Rigid Motions . . . 38
  3.5.1 Dual Quaternions . . . 38
  3.5.2 Axis-angle and Translation Vectors . . . 44
3.6 Comparison of Gaussian Process Variants . . . 44
3.7 Experimental Evaluation . . . 46
  3.7.1 Numerical Evaluation . . . 46
  3.7.2 Real World Experiment I . . . 48
  3.7.3 Real World Experiment II . . . 51

3.7.4 Simulation for Comparison 53
3.8 Discussion 55

4 Human Decision Process Estimation 57
4.1 Related Work and Open Problems 58
4.2 Differentiation of Potential Movement Plans 59
4.2.1 Data Specifications and Clustering Algorithms 59
4.2.2 Generalization of GMM Clustering to Rigid Motions 61
4.3 Probabilistic Classification of Desired Trajectory 64
4.3.1 Similarity Measure and Class Probabilities 65
4.3.2 Classification Rules 67
4.4 Online Trajectory Generation including Sample Leveraging 67
4.4.1 Sample Leveraging 68
4.4.2 Initialization 69
4.4.3 Update Step 70
4.4.4 Characteristic Properties and Application 71
4.5 Experimental Evaluation 72
4.5.1 Generalized Online GP Demonstration 74
4.5.2 Rigid Motion Clustering of Shoe Receiver 77
4.5.3 Classification Analysis in Comparison with Humans 79
4.6 Discussion 86

5 Ambient Symptom Recognition in Parkinson's Disease 89
5.1 Background Information on Parkinson's Disease 89
5.1.1 Parkinson's Disease Symptom Rating 90
5.1.2 Related Work and Open Problems 90
5.2 Data Collection 92
5.3 PD Symptom Recognition 93
5.3.1 Data Analysis 94
5.3.2 Monitoring Unit: Data Processing and Feature Generation 96
5.3.3 Fault Detection Unit: Multi-layer GP Model 99
5.3.4 GP Hyperparameter Characteristics 101
5.4 Statistical Analysis 101
5.4.1 Feature Vector Reduction 101
5.4.2 Disjointed Patient Sets 102
5.4.3 Non-uniform Symptom Distribution 102
5.4.4 Initial Hyperparameters 103
5.4.5 Training Accuracy of Symptom Severity 103
5.5 Results 104
5.5.1 Individual and Total GP Model Accuracies 104
5.5.2 Interpretation 107
5.6 Discussion 108

6 Conclusion and Future Directions 109

List of Figures 113

List of Tables **119**

Bibliography **121**

Notation

Acronyms and Abbreviations

GP	Gaussian process
PD	Parkinson's disease
HMM	Hidden Markov model
GMM	Gaussian mixture model
DL	Deep learning
CNN	Convolutional neural network
GPDM	Gaussian process dynamical model
SID	System identification
DoF	Degrees of freedom
RMSE	Rooted mean square error
MDS-UPDRS	Movement Disorder Society - Unified Parkinson's Disease Rating Scale
IMU	Inertial measurement unit
SVM	Support vector machine
KRLS	Kernel recursive least-squares
kNN	k-nearest neighbor
RGB-D	red, green, blue and depth
DBSCAN	Density-based spatial clustering of applications with noise
HAC	Hierarchical agglomerative clustering
EM	Expectation-maximization
BIC	Bayesian information criterion

UK	United Kingdom
AIMS	Abnormal involuntary movement scale
BK	Bradykinesia
DK	Dyskinesia
TM	Tremor
PSD	Power spectral density
LOSO	Leave-one-subject-out
FP	False positives
FN	False negatives

Symbols

Groups, fields, rings and *(vector) spaces* are denoted by capital letters using a blackboard bold notation (if existent). *Indices* and *scalar variables* are denoted by lower case italic Latin-script or Greek letters, while *vector-valued variables* are denoted by lower case bold Latin-script or Greek letters. *Matrices* are denoted by capital bold letters. *Sets* are denoted by capital script or italic letters and *functions* are denoted by lower case italic letters including a subscripted label, except widely used other convention are available.

Spaces, Groups and Algebras

$\mathbb{N}$	natural numbers
$\mathbb{R}^n, n \in \mathbb{N}$	n-dimensional Euclidean space
$\mathbb{R}_0^+, \mathbb{R}^+$	non-negative real numbers, positive real numbers
$SE(3)$	special Euclidean group
$TSE(3)$	tangent bundle
$T_{\mathfrak{g}}SE(3)$	tangent space to the tangent point $\mathfrak{g}$
$TS_{\mathbf{q}}$	tangent space to the unit sphere S_3 at tangent point $\mathbf{q}$
$\mathbb{H}$	skew field of quaternions
$\mathbb{H}_D$	ring of dual quaternions
$SO(3)$	special orthogonal group
$\mathfrak{so}(3)$	Lie algebra to the Lie group of rotations $SO(3)$
$\mathfrak{se}(3)$	Lie algebra to the Lie group of rigid motions $SE(3)$
$\{\pm 1\}$	normal subgroup

Operators

$\\| \cdot \\|$	norm
$\| \cdot \|$	absolute value
$\cdot^\top$	transpose
$\langle \cdot, \cdot \rangle$	dot product
$(\cdot, \cdot)$	tuple or pair
$\bar{\cdot}$	conjugate
$\circ$	composition
$\sim$	distributed as
$\simeq$	homeoporphic to
$\times$	direct product of spaces
$\times$	cross product of vectors
$\rtimes$	semi-direct product
$\wedge$	logical and
$\vee$	logical or
$\equiv$	equivalent
$\setminus$	set minus
$/$	quotient of sets, i.e. $S_3/\{\pm 1\}$ orbit space to the equivalence class $\{\pm 1\}$
$\sqrt{\cdot}$	square root
$\perp, \not\perp$	perpendicular, not perpendicular
$\operatorname{diag}(\cdot)$	diagonal elements of a matrix
iff	if and only if
$\cdot\big\|_{\therefore}$	set or function $\cdot$ is restricted to $\therefore$
$\hat{\cdot}$	estimated value for variable $\cdot$
$(\cdot)_{-i}$	remove operator of the i^{th} element/row in a vector/matrix
$(\cdot)_{-i,-i}$	remove operator of both, the i^{th} row and column, in a matrix
$[\cdot]_\times$	skew-symmetric matrix operator corresponding to vector $\cdot$
$\#$	cardinality of a set
$\mathbb{B}(\cdot)$	boolean operator
$\lfloor \cdot \rceil$	round operator with constraint to values $\{0, 1, 2, 3, 4\}$

Functions

$\tilde{f}_M(\cdot)$	noisy human motion execution dynamics
$\tilde{f}_{M_\mathrm{d}}(\cdot)$	noisy motion execution dynamics with fixed desired trajectory
$f_{M_\mathrm{d}}(\cdot)$	noise-free motion execution dynamics with fixed desired trajectory
$\tilde{f}_D(\cdot)$	leveraged noisy human decision dynamics
$f_D(\cdot)$	measurable system output encoding probabilities
$\tilde{f}_S(\cdot)$	noisy internal motor state dynamics
$\tilde{g}_S(\cdot)$	measurable system output encoding motor dysfunctions
$\check{f}_D(\cdot), \check{f}_S(\cdot)$	subsystem dynamics models
$g_f(\cdot), \tilde{g}_f(\cdot)$	observer function w/ and w/o Markovian property
$g(\cdot), g_1(\cdot), g_2(\cdot)$	auxiliary functions
$\varphi(\cdot)$	a real-valued stochastic process
$f = (\varphi_1, \ldots, \varphi_n)^\top$	vector-valued function
$d(\cdot, \cdot)$	distance function
$d_\mathrm{eu}(\cdot, \cdot)$	Euclidean distance
$d_\mathrm{norm}(\cdot, \cdot)$	quaternion norm metric
$d_\mathrm{arc}(\cdot, \cdot)$	arc metric
$d_{TS_\mathbf{q}}(\cdot, \cdot)$	tangent space metric
$d_\mathrm{mag}(\cdot, \cdot)$	transformation magnitude measure
$d_\mathrm{geo}(\cdot, \cdot)$	geodesic metric
$d_\mathrm{wmag}(\cdot, \cdot)$	weighted transformation magnitude measure
$d_\mathrm{maha}(\cdot, \cdot)$	generalized Mahalanobis distance to input on $SE(3)$
$m(\cdot)$	mean function
$k(\cdot, \cdot)$	kernel function
$k_\mathrm{eu}(\cdot, \cdot)$	squared exponential kernel in Euclidean space
$k_\mathrm{dot}(\cdot, \cdot)$	dot product kernel
$k_\mathrm{se:arc}(\cdot, \cdot)$	squared exponential kernel over unit quaternions with arc metric
$k_\mathrm{se:norm}(\cdot, \cdot)$	sq. exp. kernel over unit quaternions with quaternion norm
$k_{\mathrm{se}:TS_\mathbf{q}}(\cdot, \cdot)$	sq. exp. kernel over unit quaternions with tangent space metric
$k_\mathrm{se:mag}(\cdot, \cdot)$	sq. exp. kernel over dual quaternions
$k_\mathrm{se:wmag}(\cdot, \cdot)$	sq. exp. kernel over axis-angle and translation vector

$\mathbf{n}_{\mathfrak{g}}$	noise process in tangent space
$D_{\mathsf{KL}}(\cdot\|\|\cdot)$	Kullback-Leibler divergence
p	probability measure
δ_{ij}	Dirac delta function
$\tilde{\delta}_{ij}$	modified Dirac delta function
$\mathcal{N}(\cdot)$	(multivariate) Gaussian and projected Gaussian distribution
P, Q	probability distributions
$\mathbb{E}[\cdot]$	expected value
$\Pi(\cdot)$	central projection
$\exp(\cdot)$	exponential function
$\ln(\cdot)$	natural logarithm
$\mathfrak{Re}(\cdot)$	real part of a (hyper-)complex number
$\mathcal{O}$	Big O notation for complexity

Indices

κ	index of discrete time dynamics, $\kappa \geq 0$
l	index running over dimensions
i, j, k	indices

Variables

μ	mean
σ	variance
ϵ	dual unit
θ	rotation angle
n	dimension
ν	sample set size
η	number of trajectories
ϱ	cluster size
ζ	sigma points for unscented transform
$\varkappa$	scaling parameter
$\lambda, \sigma_f, \sigma_n$	length-scale, signal variance and signal noise
r	radius

ξ	symptom severity level
ϖ_3	Daubechies wavelets of order 3
ϕ, ψ, ϑ	Euler angles: “roll, pitch and yaw”
i_x, i_y, i_z	imaginary components
$c, \tilde{c}, c_{\text{scale}}, c_{\max}, \varepsilon \in \mathbb{R}^+$	constants
t, t', T	time instances
$\mathbf{s}_{\nu+1}, \gamma^2_{\nu+1}, \boldsymbol{\varsigma}_{\nu+1}, \hat{\sigma}^2_{f_{\nu+1}}$	auxiliary variables in the generalized online GP
$\beta \in \mathbb{N}^+$	dictionary budget, i.e. maximal cardinality of training data
$\mathbf{1} = (1, \dots, 1)^\top$	neutral element in respect of multiplication in vector space
$\mathbf{0} = (0, \dots, 0)^\top$	neutral element in respect of addition in vector space
$\gamma \in (0, 1]$	forgetting factor
$\rho_i, i \in \mathbb{N}^+$	weighting factors
$\chi_i, i \in \mathbb{N}^+$	mixing coefficients
$a, \tilde{a} > 0$	threshold
w_κ	unknown disturbance
v_κ	noise
$\mathrm{n}(t), \tilde{\mathrm{n}}(t)$	noise signals
$\mathbf{c} \in \mathbb{R}^\nu$	ν-dimensional vector
$\mathbf{t}$	one minute time window
$\mathbf{q}$	quaternion
$\boldsymbol{\mathfrak{g}}$	group element of $SE(3)$, i.e. a rigid motion
$\mathsf{w}^1(t), \dots, \mathsf{w}^\varrho(t)$	vector of potential movement weights
$\mathbf{h}$	vector of hyperparameters
$\mathbf{b}_{\boldsymbol{\mathfrak{g}}}$	vector of sample leverage at $\boldsymbol{\mathfrak{g}}$
$\mathbf{p}_{\boldsymbol{\mathfrak{g}}}$	vector of probabilities at $\boldsymbol{\mathfrak{g}}$
$\mathbf{z}$	feature vector obtained from inertial signals
$\mathbf{v} = (v_x, v_y, v_z)^\top$	Euclidean vector
$\mathbf{u} = (u_x, u_y, u_z)^\top$	Euler rotation axis (pseudovector)
$\boldsymbol{\tau} = (\tau_x, \tau_y, \tau_z)^\top$	vector of linear velocities
$\boldsymbol{\alpha} = (\alpha_x, \alpha_y, \alpha_z)^\top$	vector of linear accelerations
$\boldsymbol{\omega} = (\omega_\phi, \omega_\psi, \omega_\vartheta)^\top$	vector of angular velocities
$\boldsymbol{\mathfrak{s}}_{\text{acc}}, \boldsymbol{\mathfrak{s}}_{\text{gyr}}$	norm of filtered accelerometer and gyroscope signals
$\tilde{\boldsymbol{\mathfrak{s}}}_{\text{acc}}, \tilde{\boldsymbol{\mathfrak{s}}}_{\text{gyr}}$	wavelet decomposed signal

$\boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\chi}$	(generalized) GMM parameters
$\mathbf{P}$	$(\nu \times \varrho)$-dimensional matrix of responsibilities
$\mathbf{R}$	rotation matrix
$\Omega = [\boldsymbol{\omega}]_\times$	skew symmetric matrix of angular velocity vector $\boldsymbol{\omega}$
$\mathbf{M}_{\mathrm{a:corr}}$	auto-correlation matrix of unit quaternions
$\mathbf{K}$	Gram matrix
$\mathbf{S} = \mathbf{K}^{-1}$	inverse of the Gram matrix

Sets

M	Riemannian manifold
$\mathbf{B} = \{\boldsymbol{\beta}_1,\ \boldsymbol{\beta}_2,\ \boldsymbol{\beta}_3\}$	Basis of a 3D space in $\mathbb{R}^4$
$S_n, n \in \mathbb{N}$	unit n-sphere
$B_\pi(0)$	solid ball with radius $0 \le r \le \pi$ around origin
$\tilde{B}_\pi(0)$	unique parametrization set of rotation group $SO(3)$ by axis-angle
$\mathcal{D}_\nu$	sample set consisting of ν input-output pairs
$\widetilde{\mathcal{D}}_\nu$	set of rigid motion trajectories
$\check{\mathcal{D}}_\nu$	set of weighted training samples
$\mathcal{C}_i$	i-th cluster of samples
$\dot{\cup}_{i=1}^{\varrho}\mathcal{C}_i$	disjoint set of clusters
$\mathcal{X}$	index set
$\mathcal{V}$	multidimensional target set of $\check{f}$
$\mathfrak{x} = \{\,\mathfrak{g}(t)\}_{t\ge 0}$	rigid motion trajectory in $SE(3)$, i.e. sequence of rigid motions
$\{\,\mathfrak{g}_{\mathsf{d}}(t)\}_{t\ge 0}$	desired rigid motion trajectory
$\mathcal{Z}_\kappa$	measurement set at κ
$\mathcal{F}$	set of trigonometric functions
Θ	set of low level model parameters
$\mathcal{J}_{\mathbf{t}}$	sample set captured during time window $\mathbf{t}$
$\mathcal{J}_{\mathsf{TM}}, \mathcal{J}_{\mathsf{DK}}, \mathcal{J}_{\mathsf{BK}}$	index sets
$\mathcal{T}$	set of time windows $\mathbf{t}_i$
$\{\,\mathbf{x}_\kappa\}_{\kappa=1}^{\nu}$	vectorial training input set in Euclidean space
$\{y_\kappa\}_{\kappa=1}^{\nu}$	scalar training output set in Euclidean space

1

Introduction

Understanding the human has been an active research field for centuries in numerous disciplines such as psychology, medicine and biology. A wide variety of approaches to explain the human have been investigated, e.g. modeling the human brain, the nervous system or human movement generation involving muscle and joint activation, just to name a few examples. Among the many approaches to explain human functioning, we are interested in the sensorimotor process, which defines how the nervous system interacts with other body parts and the environment to produce purposeful, coordinated movements.

The sensorimotor system is required for a wide range of application areas, reaching from human-robot interaction (HRI) over gaming and mobility to healthcare. In gaming applications, for instance, the movement behavior prediction allows for pre-rendering of a computer simulated avatar, controlled by the human player, resulting in smooth movement visualization. In mobility, the accurate prediction of human action is essential to avoid accidents with (semi-)autonomous cars.

In this thesis we focus on the application in human-robot cooperation and healthcare.

Human-Robot Cooperation

The probably most prominent application field of human motion prediction is HRI, e.g. jointly carrying a table. According to [1], the prediction of human motor behavior is a prerequisite for successful and intuitive human-robot interaction. We believe that the robot on top needs to be able to adapt to human cooperation partners, which requires decision making for the adequate behavior. Hence, the appropriate interpretation of human motor intent is crucial for *seamless human-robot cooperation*, which is an extension of classical physical HRI: It additionally comprises the appropriately reaction and adaptation of the robot to the human intent (e.g. stepping aside if the human wants to pass).

Healthcare

In healthcare the estimation of human motor behavior is essential for fault detection of the human sensorimotor system, such as characteristic motor abnormalities. Malfunctioning of the sensorimotor system indicate movement-related deficits, which may be typical symptoms exhibited in neuro-degenerative diseases such as Parkinson's or Huntington's. Being able to detect and interpret the sensorimotor system fault allows for symptom recognition and estimation, and thus, enables appropriate patient medication.

In order to generate skilled and efficient actions, the sensorimotor system must find solutions to several problems inherent in sensorimotor control, including nonlinearity, nonstationarity, delays, redundancy, uncertainty, and noise [2]. Further, the human sensorimotor system is a system, that acts in a dynamically changing environment. Hence, understanding the human motor system is challenging. An overview over the unsolved issues about how the human sensorimotor system generates skillful motor behavior is provided in the following.

It is unknown, for instance, how humans select a particular solution out of an infinite pool of possible movements to achieve a certain task goal. The various types of noise included in the human nervous system, spanning from perception over planning to kinematics, were thoroughly investigated (e.g. see [3–5]). Obviously, the human brain is able to manage the noise, when the human generates skillful movements. However, we do not fully understand how the sensorimotor control copes with the noise contamination [2]. Further challenges in the motor system concern the internal and external uncertainty. On top of the noisy human sensors [6], unknown environmental properties add to the environmental uncertainty [7, 8]. Motor commands, descending from brain to endpoint movement, undergo a nonlinear transformation, which results in complex muscle activation patterns including further uncertainty [9]. Moreover, the multi-joint, multi-link structure of the human skeleton exhibits complex nonlinear dynamics [10] and the transformation from a command in the posterial parietal cortex to an executed muscle contraction takes up to 40 ms according to [2], depending on the length and type of nerve fiber. As motor commands can hence be out of date, the motor control must consider delay times, which otherwise could lead to instability in the motor behavior. There exists no integrated model of the sensorimotor process, which could cope with all challenges. How the human nonetheless is able to produce well coordinated limb movement, is hence unsolved.

So far, there is no general model available for the human sensorimotor system to generate goal-directed movements, as the sensorimotor system dynamics is complex and evolves highly inconsistent.

1.1 Challenges

In this thesis we develop an integrated framework for estimating and predicting human motor behavior. In the following, we explain the main challenges that arise from the complexity of the human sensorimotor system and briefly outline how we cope with those issues.

Challenge 1.1.1 *Traditional methods are insufficient to describe the full human motor system, as the human exhibits unknown and nonlinear system dynamics that cannot be captured with traditional methods. All available parametric models can at most partially describe the complex dynamics of human motor behavior, e.g. in a restricted setting as for completing a reaching task.*

In complex task settings there is no model based on first-order physical principles that suffices to predict the human motor behavior [11]. We suggest to employ a data-driven modeling approach for appropriate consideration of unconstrained movement behavior and actions of humans.

Challenge 1.1.2 *Human movement naturally evolves on the space of rigid motions, as human motions include rotations and translations. So far, however, the existing data-driven* *methods cannot handle dynamics evolving in the space of rigid motions, as rotations are not appropriately considered.*

We introduce a rigorous mathematical framework for modeling human motor behavior dynamics. The framework comprises an estimation of the human motor state employing multiple GPs and a generalization of the valid input domain of the GP to full rigid motions, namely the special Euclidean group $SE(3)$. Therefore, the kernel function inside the GP are modified to exploit properties of the input data on the rigid motion manifold.

Challenge 1.1.3 *Human behavior in unconstrained environment is highly variable. Given a task allows for free movements, humans behave differently in multiple repetitions of the same task. The variability in the behavior is due to uncertainty in the internal decision making and motor behavior of humans.*

We consider the uncertainty in human decision making and adaptation to the environment in the human movement generation through probabilistic rigid motion trajectory classification. The correctness and accuracy of all aspects of our approach is validated on real human motion data.

We structure the reminder of this chapter as follows. We provide an overview of related work on human sensorimotor process modeling and review data-driven methods, as they are considered suitable for modeling the human. Subsequently, we present the contribution of this thesis.

1.2 Human Motor Behavior

In literature, there are various computational methods available to model certain aspects of human motor behavior. Various studies suggest that the sensorimotor control system involves Bayesian decision theory [12–14]. In [12], for instance, the subjects learn a posterior distribution of time delay, in parallel estimated a likelihood for the reliability of visual information and involved probabilistic reasoning to combine those signals. Optimal feedback control (OFC) partially explains how the task execution is selected by solving the redundancy of the motor system, and therefore, is successfully applied to trajectory prediction [15, 16]. OFC finds the best possible feedback control law for a given task that minimizes a mixed cost function with components that specify both accuracy and energetic costs. Therefore, an important feature is that it only corrects for deviations that are task relevant and allows variation in task-irrelevant deviations, following the so-called minimum intervention principles [2]. Additionally, simple movement prediction can be carried out using the minimum jerk profile for point-to-point movements [17, 18], although certain movements have some path curvature depending on gravitational constraints [19]. Besides, the studies of [5, 20, 21] suggest the involvement of impedance control in the sensorimotor control system. Impedance control can be employed to describe the muscle intrinsic properties (stiffness and damping), and thus, is suitable for approximating feedback delays and uncertainty. Conclusive evidence for a predictive control or forward models in the human motor system could so far not be provided, as the predicted event is not a measurable output. Numerous studies, however, suggest their involvement [22–25].

Even though, the review of computational methods may partially explain the human motor system functioning, the neurophysiological implementation of the presented methods is not understood and thus, no integrated framework for the human sensorimotor system is available [2]. If regarded separately, none of the previously presented methods considers all aspects required for holistic human motor behavior modeling.

The most promising approximation are achieved employing learning approaches. Data-driven modeling is regarded superior for modeling human motor behavior due to its capability to represent complex dynamical systems [26]. This might be due to the human employing learning in the own sensorimotor system. According to [27], the human optimizes his/her motions through the redundancy in the degrees of freedom of the human body. This induces strongly nonlinear motions with high variance due to environmental constraints. In [28] it is shown that such nonlinearities in the muscles are learned error-based. Additionally, humans learn how to adapt the neuromuscular system to changes in the environment [29]. For instance, the kinematic parameters of reaching movements are tuned by object characteristics such as shape [30] and mass distribution [31], as well as its location [32] and orientation [33]. These object characteristics are integrated into human motor behavior in order to ensure that the subsequent motor command is optimized at a critical final phase of the task [34], as proposed in a theoretical framework termed end-state comfort.

1.3 Data-Driven Modeling

Data-driven modeling is a technique, where the parameters and components of a model are determined based on the characteristics of some set of observations of an unknown system, such that the model best describes the observations. The possible application areas for data-driven modeling techniques are numerous. They comprise the modeling of complex dynamical systems such as soft robots, unknown object estimation and mechanical system modeling just to mention a few examples. Additionally, in image data processing learning approaches were successfully applied, and in traffic flow forecast a data-driven nonparametric approach is advantageous according to [35]. In the present context, we aim to apply data-driven modeling to predict human motor behavior.

Probably the most widely studied data-driven techniques for modeling the human motor system dynamics are Gaussian processes (GPs), Gaussian mixture models (GMMs) and Hidden Markov models (HMMs); applied for instance by [36], [37] and [38], respectively. These approaches convince owing to their characteristic to describe system dynamics from mere observations. While HMMs are particularly popular for describing motion primitives, they are limited to the discrete state space. GPs and GMMs are defined on continuous space and therefore suitable for modeling dynamics in continuous state space. Until the late 1990s, deep learning (DL) approaches were considered unsuitable for dynamical system modeling: either the dynamics are not robust to noise, or the gradients vanish [39, 40]. The author of [40] suggests to employ recurrent neural networks (RNNs) to overcome the learning long time lag problem. For complex dynamics, however, the training of deep RNNs is difficult and time consuming [41, Chap. 10]. Only recently a deep convolutional neural network (CNN) architecture was proposed that overcomes scalability and robustness issues [42]. The authors propose to use multiple convolution layers to separately store the long-term and short-term dynamics in a CNN model. In their approach, however, the CNN stores the full dynamics history in the model output and thus does not learn the dynamic mapping from one state to the next.

We focus on the GP model for its property to account for the uncertainty in each prediction besides providing a best estimator [43]. This uncertainty estimate is highly valuable in safety-critical applications such as human-robot interaction [44].

1.4 Main Contribution and Outline of the Thesis

This thesis introduces an integrated framework for human motor behavior prediction to cover a large proportion of the human sensorimotor processes including the consideration of dynamics evolving on the rigid motion manifold. We detail the framework and provide the structure of this thesis in the following.

We start by developing a holistic dynamical model employing various GPs, which is suitable to cover all relevant aspects of human motor behavior prediction (Chapter 2). The dynamics comprise three main components that correspond to rigid motion execution, decision making and motor state estimation. In Chapter 3, we detail the rigid motion prediction. We introduce a mathematically firm GP generalization for modeling rigid motion dynamics on $SE(3)$. Herein, we explain the procedure for two possible parametizations of 6-DoF rigid motions: the dual quaternions, and the axis-angle and translation vector. Chapter 4 concerns the human decision making, which consists of a parallel process of potential trajectory planning and the weighting of those optional movements to determine a certain movement trajectory. Therefore, we develop a recursive version of the GP over rigid motions for online applications, that moreover allows for weighting the training data according to its significance and information content. The overall motor state of the human is estimated in Chapter 5. We introduce a fault detection algorithm to detect and estimate motor symptoms in Parkinson's disease using a multi-layer GP. We conclude the thesis in Chapter 6.

The major contribution of each of the model parts is detailed in the reminder of this section. At the beginning of each chapter, the relevant related work and open problems are briefly reviewed. At the end of each chapter, we shortly conclude on our solution to the specific challenge.

Chapter 3: Gaussian Process for Rigid Motion Dynamics on $SE(3)$

As human movement behavior in unconstrained environments manifests as in the space of rigid motion, we provide a formal GP definition and then generalize the GP to the manifold of rigid motions. We introduce GP kernel functions on the rigid motion using the parametrization unit and dual quaternions. In addition, we propose the representation of rigid motions by axis-angle and Euclidean vectors to develop a GP model, that generalizes to non-Euclidean input space more computationally efficient. We present novel mean and kernel functions and provide proofs for both parametrizations for the important generalized squared exponential kernel to be valid. By default, the model uncertainty of the generalized GP is encoded in a Gaussian defined in the gradient field to $SE(3)$. Thus, we complete the framework by developing a proper integration method of the uncertainty prediction into $SE(3)$. We therefore define a probability distribution on $SE(3)$, which possesses the properties of the Gaussian bell curve. Our generalized GP model is validated on simulated and real experimental data. In a regression performance analysis, synthetically generated dynamics are modeled by the GP over dual quaternions, and are evaluated in comparison to the traditional GP. In an experimental comparison both GP generalizations are evaluated in terms of efficiency and accuracy. The evaluations confirm that the introduced framework for GP modeling over rigid motions significantly outperforms the traditional way of including orientation into the GP via Euler angles. These results were preliminary published on conferences [45–47] and as journal papers [48, 49].

Chapter 4: Human Decision Process Estimation

Human motor behavior includes the adaptation to other humans or the environment, which involves the decision making for a specific movement. We model the decision making by a parallel trajectory planning and decision weighting algorithm. To allow for online motion prediction, a novel recursive generalized GP model for rigid motion dynamics is introduced. For modeling the decision process a clustering algorithm that enhances the GMM cluster fitting to input data in $SE(3)$ is presented. The resulting motion clusters are used as potential movement trajectories in the human internal decision process. To calculate the probability distribution for an unseen rigid motion to belong to any of the trajectory clusters, we employ a novel Mahalanobis measure. It utilizes the findings from the generalized GP model framework to allow for measuring distance on $SE(3)$ in accordance to the Mahalanobis concept. The probabilities of the data clusters are used to weight the training samples that are assigned to the respective cluster. We then employ the weighted training data in the generalized online GP to predict the human decision process. The experimental evaluation of our approach to model the decision process is conducted on a toy example, namely lower limb assistance. More precisely, an assistant robot puts on a shoe to a human receiver. The receiver performs foot movements towards freely selected shoe dressing poses, which are clustered employing the generalized GMM clustering algorithm. We demonstrate that in the specific toy example, five is the optimal number of trajectory clusters. The presented trajectory classification method is evaluated in comparison to human classification abilities and shows faster and more accurate results than human subjects achieve in a motion classification task from video data. Additionally, we evaluate the performance of the recursive GP model in an experiment where it is used for online motion estimation in an object handover task using real data. The results were partially published in [48, 50].

Chapter 5: Ambient Symptom Recognition in Parkinson's Disease

We estimate hardly measurable internal human motor states exemplary for Parkinson's disease (PD) patients. We provide an approach that not only autonomously detects each of the PD motor symptoms, i.e. bradykinesia with and without tremor and dyskinesia, but moreover estimates the symptoms' severity. Further, we do not require the patient to perform specific movement tasks, but complete ambient symptom recognition during unconstrained free living activities. Finally, we achieve reliable results from a single commercially available wrist-worn low-cost wearable sensor. The key to this remarkable performance comes from three aspects: (1) the identification of appropriate data features for the PD patients' motor symptom detection and symptom severity estimation, (2) our approach to model PD symptoms by a dynamical system and (3) the multi-layer structure of our method. Specifically, we consider the PD motor symptom evolution as a dynamical system that can be described by characteristic movement features and the patient's motor states and symptoms. Then, we propose to estimate the unknown function in the dynamical system by a multi-layer approach in which we employ multiple GP models. As the tremor motion is highly characteristic and thus, clearly differentiable from other motions, the model estimates the tremor presence and its severity in the first layer, based on the frequency ratio of the tremor motion versus voluntary actions. Subsequently, in the second layer, the remaining data are used to analyze the severity of dyskinesia and bradykinesia (without tremor). As the balanced condition is defined by the absence of movement dysfunctions, it is recognized if the severity prediction is below a certain threshold. For each of the three estimates we apply GP regression or prediction and thus, obtain a multi-layer GP model. These results are published as [51], and a second publication on this topic is in preparation.

2

Dynamics Model of the Sensorimotor System

The human sensorimotor system comprises two aspects, namely the motor system of the human, i.e. the part of the central nervous system that is involved with motion, and the human action, i.e. the movement output of human body parts after the sensorimotor signals are transformed to muscle activation and coordination. Hence, it generates the required motor commands to execute a goal-directed movement from sensory signals, such as visual information [52]. To develop a plant model of the system, a general understanding of the human sensorimotor processes is required. Multiple human-internal processes such as sensory processing, memory access and motion generation are controlled by the cerebral cortex. We consider only the relevant aspects for modeling the human motor system and human action. The sensorimotor processes comprise the cerebral components concerned with the motor action planning, decision making and motor command generation. Additionally, they comprise the motor control for generating coordinated muscle activation and limb movement to execute some action. Thus, they are responsible for the whole control loop from receiving a task goal input to outputting a goal-directed movement to achieve the task goal. A schematic visualization of the dynamics model is provided in Figure 2.1. The red dashed line divides the processes taking place in the human brain and the processes involving other body parts.

Plant Model Description

We introduce in the following our suggested plant model for the sensorimotor process describing human motor dynamics. The model incorporates three major subsystems, the decision making, the motor execution and the internal motor system state, which are highlighted by blue boxes in the figure.

Recent neurophysiological studies suggest that the human intended movements comprise two stages [53]: firstly, a parallel process of potential movement planning and weighting of the movement options to decide for one of them, and secondly the actual execution of the motion. In the plant model those stages correspond to the blue boxes 2 and 1.

In contrast to the traditional serial model in which decision making occurs before action planning, [54, 55] show that potential spatial plans for movements (to possibly multiple target locations) are represented simultaneously. Specifically, given a task goal in form of a target location $\mathfrak{g}_\mathsf{T}$, the *potential movement planner* generates multiple potential desired trajectories $\mathfrak{g}_\mathsf{d}^1(t), \dots, \mathfrak{g}_\mathsf{d}^\varrho(t)$ in parallel, which could all serve to achieve the task goal. It is not known how many potential movements $\varrho \in \mathbb{N}^+$ are generated in parallel; presumably ϱ

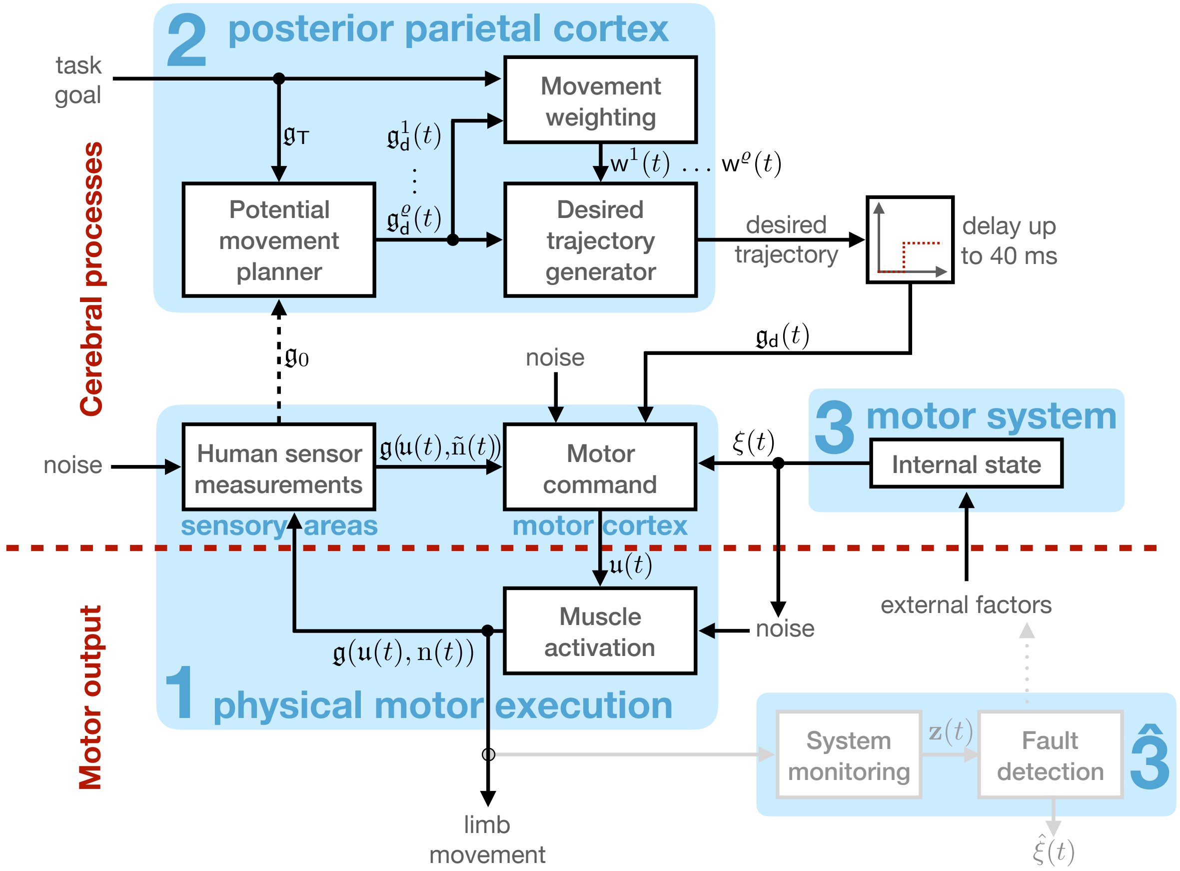


Figure 2.1: Control scheme of relevant aspects of the human sensorimotor system and motor output. Subsystem 2 concerns the decision making for a spatial movement action and takes place in the posterior parietal cortex of the human brain. Subsystem 1 concerns the subsequent movement execution of the planned action. The transformation from the desired trajectory to the actual end-point movement can have a delay of up to 40 ms. Under certain circumstances the motor control can be impaired, namely, when the internal state of the motor system is dysfunctional. Subsystem 3 models the state of the motor system. The last remaining subsystem $\hat{3}$ is an external system monitoring the human motor behavior to estimate the human internal motor system state.

depends on the task. During the process of making the decision, the potential spatial plans are balanced against each other (in our plant model this happens in the *movement weighting unit*) resulting in some weighting of the movement options $\mathrm{w}^1(t),\dots,\mathrm{w}^\varrho(t)$ [56]. In the following unit, namely the *desired trajectory generator*, a desired movement plans $\mathfrak{g}_\mathrm{d}(t)$ is generated from the weighted potential movement plans. The chosen desired trajectory $\mathfrak{g}_\mathrm{d}(t)$ is passed to the following subsystem. We name the whole parallel process that we just described *decision process*. It takes place primarily in the posterior parietal cortex [53], which in the schematic plant model is highlighted as subsystem 2.

Subsequently, the actual motion execution is initialized: By demonstrating that delay times in an online control task incur in the sensorimotor control loop, it is shown in [57] that *physical motor execution* is the temporal successor of the decision process. The motion execution is visualized in the subsystem 1. It involves sensory and motor cortical areas as well as the interaction with physical body parts including tendons, muscles and skeletal joints. The desired trajectory $\mathfrak{g}_\mathrm{d}(t)$ arrives in the *motor command unit*, which acts as a controller to generate the control variable $\mathfrak{u}(t)$ including noise. In fact, the latency of up to 40 ms is the aggregated delay from the posterior parietal cortex down to the muscle contraction [2]. The human can cope well with these delay times in the motor cortex, and generates generally suitable motor commands which are not outdated by the time they arrive at the executing body part (presumably using predictive models [58]). Therefore, we model the delay to occur before the signal arrives in the motor cortex. However, we include noise ins subsystem 1 to model the uncertainty the motor command unit has to manage. At this point in the control loop, the signal $\mathfrak{u}(t)$ exits in the human brain and traverses down the nervous system to communicate with other relevant body parts such as muscles. Through the *muscle activation* it achieves the task goal via appropriate limb movement, which generates the trajectory $\mathfrak{g}(\mathfrak{u}(t), \mathrm{n}(t))$. The muscle activation and coordination likewise includes noise. We will explain the various types of noise contained in the motor execution subsystem later in more detail. For now, we only want to point out, that those noises differ from each other. The human monitors its own motor execution using sensors such as vision and touch in the *human sensor measurement unit*, which again generates noisy measurements $\mathfrak{g}(\mathfrak{u}(t), \tilde{\mathrm{n}}(t))$. Those measurements are transmitted to the postrior parietal cortex (subsystem 2) in an event triggered fashion, specifically each time the potential movement planner needs to be reinitiated.

Remark 2.0.1 *In our straightforward applications later on, no reinitiation of the decision process will be required. Therefore, we can assume without loss of generality that the human sensor measurements $\mathfrak{g}_0$ about the relevant body parts are transmitted to the decision process only once at the beginning of each task implementation. Therefore, this signal transmission is visualized by a dashed arrow from the sensor unit to the movement planner unit.*

To get a better understanding of the noise involved in human motor execution, we reflect more in detail on human motor behavior. The human motor execution consist of a consciously and willingly performed share - this is what we call intended motion - and a non-deliberately and/or unconsciously performed share - i.e. movement that occurs from sensorimotor uncertainty. Following Orban et al. [59], there are various types of sensorimotor uncertainty: *state uncertainty*, i.e. a probability distribution over possible states that result from a motor command based on noisy human sensor input, *structural uncertainty*, i.e. unknowns in motor equations that make system identification difficult, and *parametric uncertainty*, i.e. unknown dynamics settings of the body or unknown task characteristics.

Remark 2.0.2 *In our model, the state uncertainty affects the* sensor measurement unit*, the structural uncertainty affects the* motor command*, and the parametric uncertainty affects the* muscle activation*.*

Remark 2.0.3 *The decision process is assumed to be noise-free. For simplicity, we model the human system to generate "desired trajectories", which are perfect in that sense that they describe the true desire of the human and not some corrupted version of it.*

Unintentional motor behavior includes movements the human is aware of but cannot suppress, and movements the human executes unconsciously. We call them unintended human motions. Usually, the sensorimotor uncertainty manifests in marginal unintended movements such as little drift. Since most people exhibit this behavior all the time, the condition of the human motor system including marginal unintended motion is referred to as *normal or balanced motor state.* So far, we have assumed, that the human motor state is balanced and thus, noise ration in the signals is reasonably small, such that signals $\mathfrak{u}(t)$, $\mathfrak{g}(\mathfrak{u}(t), \mathrm{n}(t))$ and $\mathfrak{g}(\mathfrak{u}(t), \tilde{\mathrm{n}}(t))$ are not superimposed by the various noises. Now, we investigate the case when the human motor behavior is *unbalanced.* The occurrence of significant unintended movements is the result of specific motor abnormalities, for which the *internal state of the human motor system* $\xi(t)$ is responsible. It depends on many unknown external factors such as nutrition, excitement or medication. In Parkinson's disease (PD) for instance, the over and under supply of the cerebral cortex with dopamine results in an unbalanced internal motor state. We consider the *state of the motor system* in the model plant for the human motor dynamics in the subsystem 3. The internal motor state $\xi(t)$ directly affects the motor command such that the generated signal $\mathfrak{u}(t)$ is corrupted in the unbalanced case compared to the balanced. Additionally, the internal motor state $\xi(t)$ indirectly affects the muscle activation unit. Specifically, it changes the noise ratio in the limb movement $\mathfrak{g}(\mathfrak{u}(t), \mathrm{n}(t))$.

Plant Model Approximation

In the reminder of this chapter we detail our approaches to approximate the model subsystems $1-3$. We start with the subsystem 1, the physical motor execution. We assume a desired trajectory to be given and the internal motor state to be balanced. Hence, we approximate the closed loop dynamics of subsystem 1. This dynamics evolves on the special Euclidean group $SE(3)$, since human movements naturally are rigid motions. Thereafter, we approximate the noise-free dynamics in subsystem 2. We propose approximators for the parallel movement planning and the movement weighting to determine the desired trajectory. Finally, we present an estimator for the internal motor state dynamics in subsystem 3. Since the internal state of the motor system is hardly accessibly with non-invasive methods, we employ an external monitoring system to detect unbalanced states. Subsystem $\hat{3}$ is not part of the human dynamics model, but serves us as *fault detection system.* We infer an estimate $\hat{\xi}(t)$ for the human internal motor state from the motor execution. In future work, we aim to influence the external factors to regulate the state of the human motor system; visualized by the grey dotted arrow in Figure 2.1. We mark the elements outside the human body in the plant model through light grey shading. We apply our motor behavior fault detection to PD patients and estimate their internal motor states.

Remark 2.0.4 *The motor abnormalities induced by unbalanced internal motor states contaminate the motor execution, so that the intended motion share in the resulting limb movement becomes indistinguishable from the unintended motion share. Therefore, we will not consider dynamics of the motor execution under the influence of unbalanced internal motor states in this thesis.*

The dynamics in each of the subsystems $1-3$ of the human sensorimotor dynamics plant model are unknown. Approximating them is an open problem using traditional methods. Therefore, we propose a data-driven approximation using (generalized) GP models.

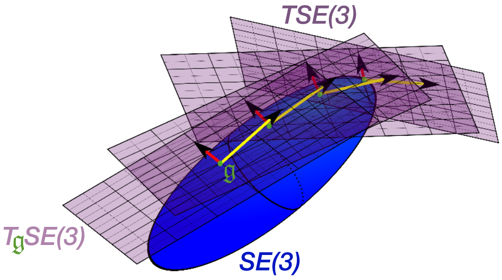


Figure 2.2: Scheme of the noisy nonlinear motion dynamics mapping from the space $\mathfrak{g} \in SE(3)$ to the corresponding velocity space $T_{\mathfrak{g}}SE(3)$.

2.1 Human Motion Dynamics on $SE(3)$

In this section we aim to approximate the dynamical subsystem of physical motor execution. We assume that this closed loop dynamics is given by $\dot{\mathfrak{g}}(t) = \tilde{f}_M(\mathfrak{g}(t), \mathbf{n}_{\mathfrak{g}}(t), \mathfrak{g}_\mathsf{d}(t))$ with a continuous function

$$\tilde{f}_M\colon SE(3) \times T_{\mathfrak{g}}SE(3) \times SE(3) \to TSE(3), \tag{2.1}$$

where the noise process $\mathbf{n}_{\mathfrak{g}}(t)$ comprises the noise in the motor command and the muscle activation. Since we assume that the human generates a finite set of desired trajectories $\mathfrak{g}_\mathsf{d}(t)$, we can re-define the dynamics $\tilde{f}_M$ to include the desired trajectory,

$$\tilde{f}_M(\mathfrak{g}(t), \mathbf{n}_{\mathfrak{g}}(t), \mathfrak{g}_\mathsf{d}(t)) = \tilde{f}_{M_\mathsf{d}}(\mathfrak{g}(t), \mathbf{n}_{\mathfrak{g}}(t)). \tag{2.2}$$

The nonlinear motion dynamics $\tilde{f}_{M_\mathsf{d}}$ describes a mapping in continuous time from the rigid motions $\mathfrak{g} \in SE(3)$ and noise $\mathbf{n}_{\mathfrak{g}}(t)$ to the corresponding velocity space $TSE(3)$, which is described by a tangent bundle

$$TSE(3) = \{(\mathfrak{g}, \dot{\mathfrak{g}}) |\, \mathfrak{g} \in SE(3),\ \dot{\mathfrak{g}} \in T_{\mathfrak{g}}SE(3)\}\,. \tag{2.3}$$

The tangent bundle (2.3) consists of pairs of a rigid motion $\mathfrak{g}$ and corresponding time derivative $\dot{\mathfrak{g}}$ in the tangent space $T_{\mathfrak{g}}SE(3)$. We further consider the noise process $\mathbf{n}_{\mathfrak{g}}$ to be defined in the tangent space $T_{\mathfrak{g}}SE(3)$, and we obtain noisy measurements of the velocity through the mapping

$$\tilde{f}_{M_\mathsf{d}}\colon SE(3) \times T_{\mathfrak{g}}SE(3) \to TSE(3). \tag{2.4}$$

Remark 2.1.1 *A noise-free version of the rigid-body motion dynamics is given by*

$$f_{M_\mathsf{d}}\colon SE(3) \to TSE(3). \tag{2.5}$$

It defines the mapping $\dot{\mathfrak{g}}(t) = f_{M_\mathsf{d}}(\mathfrak{g}(t))$.

A schematic visualization of (2.5) is provided in Figure 2.2.

Our objective is to employ GP regression for finding an approximation of the unknown dynamics (2.4). Recall, we seek to learn human motion dynamics using the GP model due to its favorable property to account for uncertainty besides providing a best estimator. Nevertheless, a major drawback of this model is that it is defined in Euclidean space. Even though a formulation with non-Euclidean input space is possible in principle, the traditional formulation requires a Euclidean vector space structure on the input as well as the output space.

Remark 2.1.2 *Likewise the GP model, any other model incorporating Gaussian distributions is defined solely in Euclidean space, due to the nature of the Gaussian probability distribution. For instance, this relates to the GMM.*

Unconstrained human motion dynamics evolve naturally in the space of rigid motions, i.e. on the special Euclidean group $SE(3)$. The possible movements, hence, consist of rotation and translation, which together generate the special Euclidean group $SE(3)$ with elements $\mathfrak{g}$, the rigid motions. For rotations, however, there exists no representation in Euclidean space. Hence, they cannot be modeled accurately using the traditional GP model. The common procedure, see e.g. [60], is to use the Euler angles representation for orientations, as close to zero their space is almost Euclidean. However, when the rotation is large, for example at high angular speed with low sampling frequency or if training data is sparse, this approach leads to inaccurate predictions. Therefore, we suggest to generalize the GP to incorporate input domains within the non-Euclidean space of rigid motions $SE(3)$. The generalization enables us to concisely learn and predict human motion dynamics such as the dynamics of subsystem 1 in our human motor behavior model in Figure 2.1. Advantageously, in the dynamics formulations (2.4) and (2.5) the output space of the GP is a tangent bundle $TSE(3)$, which provides a Euclidean vector space structure. In consequence, if we find valid mean and kernel functions over the rigid motion space $SE(3)$, the GP output in form of a Gaussian distribution is well-defined, and therefore also the GP for rigid motions. We will introduce all required generalizations in Chapter 3.

To take maximum advantage of the GP proposed uncertainty, it is additionally desirable to provide an uncertainty estimate also on the rigid motions directly. Given the dynamics input space is a Riemannian manifold M [61], we can integrate the probability distribution from the velocity space to $SE(3)$. Hence, we are able to define a generalized Gaussian process on the rigid motions.

Remark 2.1.3 *In many real world applications it is advantageous to model a dynamical system in discrete-time, e.g. when observations are received at sequential time instants only. The presented approach of approximating a dynamical system of form (2.4) is indeed analogously applicable to discrete-time dynamics in $SE(3)$. Note, however, that it is not possible to model discrete-time dynamics on $SE(3)$ using the classical formulation of a mapping from $\mathfrak{g}_\kappa$ to $\mathfrak{g}_{\kappa+1}$, as $SE(3)$ does not provide the vector space structure, which is required for the output space of the GP.*

2.2 Decision Process Dynamics

In this section we present our approximation to the dynamics of the decision process (subsystem 2 in Figure 2.1). We assume the human decision process dynamics is given by

$$\dot{\mathfrak{g}}_{\mathsf{d}}(t) = \check{f}_D(\mathfrak{g}_{\mathsf{d}}(t), \mathfrak{g}_{\mathsf{T}}), \tag{2.6}$$

where $\mathfrak{g}_{\mathsf{T}}$ denotes the task goal. Note, the drift term $\check{f}_D$ is unknown. Since we do not have access to the human internal decision making units, we use human behavior in similar situations to approximate the unknown dynamics.

We start with the potential movement planning. Hence, our first aim is to estimate the potential desired trajectories $\{\,\mathfrak{g}_{\mathsf{d}}^{1}(t)\}_{t\geq 0}\,, \dots, \{\,\mathfrak{g}_{\mathsf{d}}^{\varrho}(t)\}_{t\geq 0}$, where $\varrho \in \mathbb{N}^{+}$ is a suitable number of movement options. Therefore, we capture multiple repetitions of human movement executions, given an initial configuration $\mathfrak{g}_0$, to achieve a certain task goal $\mathfrak{g}_{\mathsf{T}}$.

In the following, we analyze how those variable realizations of each time the same task goal $\mathfrak{g}_{\mathsf{T}}$, serve as estimates for the potential desired trajectories. We assume the rigid motion planning internally to be noise free using dynamics (2.5). The executed human movements, however, are disturbed by a noise process $\mathbf{n}_{\mathfrak{g}} \in T_{\mathfrak{g}}SE(3)$ of sensorimotor uncertainty as defined in (2.4). Hence, the finite set of recorded movement trajectories $\widetilde{\mathcal{D}}_{\eta} = \{\,\mathfrak{x}_k\}_{k=1}^{\eta}$, where each trajectory $\mathfrak{x}_k = \left\{\,\mathfrak{g}^{k}(t)\right\}_{t\geq 0} \subset SE(3)$, is assumed to represent a noisy random sample from the set of desired trajectories

$$\widetilde{\mathcal{D}}_{\eta}^{\mathsf{d}} = \left\{\left\{\,\mathfrak{g}_{\mathsf{d}}^{1}(t)\right\}_{t\geq 0}, \dots, \{\,\mathfrak{g}_{\mathsf{d}}^{\varrho}(t)\}_{t\geq 0}\right\}. \tag{2.7}$$

An estimate for the unknown parameter ϱ is obtained by unsupervised clustering of the human movement trajectories $\widetilde{\mathcal{D}}_{\eta}$ into disjoint subsets $\uplus_{j=1}^{\hat{\varrho}}\mathcal{C}_j$. Each of the clusters $\mathcal{C}_j$, $j = 1, \dots, \hat{\varrho}$ contains a trajectory bundle that approximates the corresponding desired trajectory $\left\{\,\mathfrak{g}_{\mathsf{d}}^{j}(t)\right\}_{t\geq 0}$, $j = 1, \dots, \hat{\varrho}$.

Remark 2.2.1 *Only the captured rigid motion data into the clusters requires to be disjoint, the spatial distribution of the performed trajectories in various clusters is likely to be overlapping.*

Remark 2.2.2 *The motion trajectory bundles jointly represent the variety of spatial movement plans, and simultaneously the execution variability of a certain plan, which is encoded in the variance of the corresponding trajectory bundle* $\mathcal{C}_j$, $j = 1, \dots, \varrho$.

Next we provide an approximation of the potential movement weighting unit. We define a vector $\mathbf{p}_{\mathfrak{g}} \in [0,1]^{\varrho}$ encoding the probability of a rigid motion $\mathfrak{g} \in SE(3)$ to belong to any of the disjoint trajectory clusters $\mathcal{C}_1, \dots, \mathcal{C}_{\varrho}$. Hence, the probability vector is given by

$$\mathbf{p}_{\mathfrak{g}} = (p_1(\,\mathfrak{g} \in \mathcal{C}_1), \dots, p_{\varrho}(\,\mathfrak{g} \in \mathcal{C}_{\varrho}))^{\top}, \tag{2.8}$$

where $\sum_{j=1}^{\varrho} p_j = 1$. To achieve an estimation for the desired trajectory prioritization $\mathsf{w}^{1}(t), \dots, \mathsf{w}^{\varrho}(t)$, we defined $\mathbf{p}_{\mathfrak{g}}$ to be calculated by the mapping

$$\begin{array}{rccc} f_D: & SE(3) \times SE(3)^{\nu} & \rightarrow & [0,1]^{\varrho} \\ & (\,\mathfrak{g}, \uplus_{j=1}^{\varrho}\mathcal{C}_j) & \mapsto & (p_1(\,\mathfrak{g} \in \mathcal{C}_1), \dots, p_{\varrho}(\,\mathfrak{g} \in \mathcal{C}_{\varrho}))^{\top}, \end{array} \tag{2.9}$$

where $\nu = \#\uplus_{j=1}^{\varrho}\mathcal{C}_j$ is the total number of captured rigid motion samples.

Remark 2.2.3 *An intuition for the weighting process is provided by the following: At the beginning of the decision process, i.e. when the desired trajectory* $\{\,\mathfrak{g}_{d}(t)\}_{t\geq 0} = \mathfrak{g}_{d}(0) = \mathfrak{g}_0$ *only consists of the initial rigid motion, the probability vector* $\mathbf{p}_{\mathfrak{g}}$ *is uniformly distributed, i.e.* $p_j = 1/\varrho$ *for all* $j = 1, \dots, \varrho$. *As the movement generation proceeds, the probability vector* $\mathbf{p}_{\mathfrak{g}}$ *prioritizes a certain cluster, and thus, it becomes clear to which trajectory class the current motor behavior belongs.*

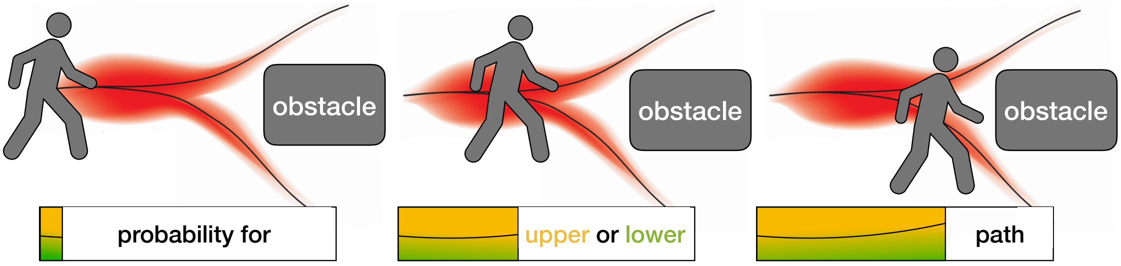


Figure 2.3: Scheme of the decision process consisting of parallel motion prediction on $SE(3)$ and estimation of the belonging class.

The functioning of the decision making unit *desired trajectory generator* in subsystem 2 of the human plant model, is not known in general [2]. We assume that the desired trajectory generator recursively operates on the potential movement planning and movement weighting. More precisely, we consider a recursive strategy to generate a desired trajectory out of the potential movement clusters $\mathcal{C}_1, \dots, \mathcal{C}_\varrho$. Hence, we assume the dynamics

$$\begin{aligned} \dot{\mathfrak{g}}(t) &= \tilde{f}_D(\mathfrak{g}(t), \mathbf{p}_{\mathfrak{g}}(t), \mathbf{n}_{\mathfrak{g}}(t)), \\ \mathbf{p}_{\mathfrak{g}}(t) &= f_D(\mathfrak{g}(t), \uplus_{j=1}^{\varrho} \mathcal{C}_j) \end{aligned} \tag{2.10}$$

to behave like the true human decision process dynamics (2.6).

The captured set of human trajectories $\widetilde{\mathcal{D}}_\eta = \{\mathfrak{x}_k\}_{k=1}^{\eta}$ is used for training a data-driven model to predict a new desired motion trajectory $\{\mathfrak{g}_\mathfrak{d}(t)\}_{t\geq 0}$. Simultaneously, we want the estimated weights (i.e. the probabilities $p_1, \dots, p_\varrho$) to influence the prediction of the desired motion trajectory. Therefore, we incorporate the weight vector $\mathbf{p}_{\mathfrak{g}}$ into our approximation of the decision process dynamics $\tilde{f}_D$. The cluster probabilities are used to leverage the data samples in every corresponding cluster, which consequently effects the proceeding movement prediction. In consequence the prioritization of one cluster over the others varies over time. Figure 2.3 illustrates the parallel process of motion prediction and cluster weighting.

Remark 2.2.4 *The recursive algorithm for the desired motion prediction is continued until one of the human movement clusters* $\mathcal{C}_1, \dots, \mathcal{C}_\varrho$ *is selected through a Dirac-like distribution of the probabilities.*

We seek to employ the generalized GP model in the decision process dynamics. Therefore, we require it to run online and to incorporate weighted training samples. We introduce a recursive version of the generalized GP and extend it to leveraged input to meet the requirements. Additionally, we present a realization for the mapping (2.9). All details on the necessary methodology will be introduced in Chapter 4.

2.3 Motor State Dynamics

In this section we present our approach to estimate the internal motor state of the human motor system. In our plant model subsystem 3 in Figure 2.1, we assume the temporal evolution of the internal motor state to be described by the dynamics

$$\begin{aligned} &\dot{\xi}(t) = \check{f}_S(\xi(t), w(t)), \\ &\xi(0) \in \mathbb{R}, \end{aligned} \tag{2.11}$$

where the function $\check{f}_S$ is unknown but smooth and the unknown state $\xi(0)$ is initialized through a random value in $\{0, \ldots, 4\}$. The derivative $\dot{\xi}(t)$ depends on the internal motor state $\xi(t)$ and an unknown disturbance process $w(t)$, representing external factors such as medication intake, nutrition, arousal level among others.

With non-invasive methods, however, the internal motor state $\xi(t)$ is hardly determinable. Therefore, we make another assumption: If the internal motor state is not balanced, this manifests in characteristic motor abnormalities with variable intensity levels. This means, we assume we can identify the internal motor state with the intensity of motor abnormalities. This assumption underlies the standard rating procedure for the internal motor state of Parkinson's patients [62].

Remark 2.3.1 *We call the characteristic motor abnormalities "(motor) symptoms". Clinically, this convention is not accurate, since the term symptom in medical science can only describe effects of some sort of disease. We, however, denominate all motor salience as symptom, even if they are caused by other factors, such as medication for instance.*

Employing the assumption concerning the symptom manifestation, we add an external monitoring system to the human plant model (visualized in light grey Figure 2.1). Since the monitoring system captures the human motor behavior with discrete-time samples only, we discretize the dynamics in the external monitoring and fault detection system to

$$\begin{aligned} \xi_{\kappa+1} &= \tilde{f}_S(\xi_\kappa, w_\kappa), \\ \xi_0 &\in \mathbb{R}. \end{aligned} \tag{2.12}$$

Now, a measurable system output

$$\mathbf{z}_\kappa = \tilde{g}_S(\xi_\kappa, v_\kappa) \tag{2.13}$$

is additionally available, which we obtain using an external tracking sensor for the human motor executions. Both functions $\tilde{f}_S, \tilde{g}_S$ are unknown but smooth. Since the internal motor state affects the motion execution, the desired trajectory of the human is corrupted and the executed movement includes the motor dysfunction in addition to the standard noise level. The monitoring unit captures the resulting human limb movement $\mathfrak{g}_\kappa$ including the patient's motor dysfunction, and transforms it to a meaningful signal $\mathbf{z}_\kappa$. More precisely, the monitoring unit generates features from the captured signal, which are suitable for the fault detection unit. Thus, the signal $\mathbf{z}_\kappa$ at time step $\kappa \geq 0$ depends on the symptom severity ξ_κ and some noise v_κ, which consists of the intended motion share (from daily living activity), the motor execution noise and the noise of the external sensor.

We aim to infer the symptom severity level ξ_κ in (2.12) using the available measurement set $\mathcal{Z}_\kappa$ at time instance $\kappa \geq 0$, where

$$\begin{aligned} \mathcal{Z}_\kappa &= \{\, \mathbf{z}_\kappa, \mathcal{Z}_{\kappa-1}\}, \\ \mathcal{Z}_0 &= \{\, \mathbf{z}_0\}. \end{aligned} \tag{2.14}$$

A fault detection unit provides the desired property. It detects the share of motor dysfunction in the motion features. Hence, an estimate for the symptom level ξ_κ, given the measurements $\mathcal{Z}_\kappa$ is obtained by

$$\hat{\xi}_\kappa = \mathbb{E}[\tilde{g}_f(\hat{\xi}_{\kappa-1})|\mathcal{Z}_\kappa], \tag{2.15}$$

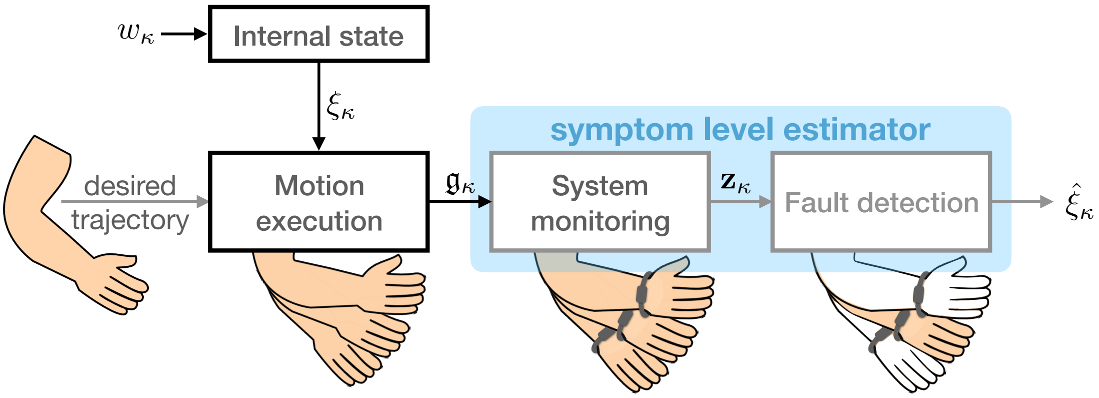


Figure 2.4: Scheme of the symptom level estimation. The internal motor state ξ_κ corrupts the intended motion in case it is unbalanced. Then, the resulting executed movement $\mathfrak{g}_\kappa$ includes motor dysfunctions. The monitoring system not only captures the signal, but also generates features $\mathbf{z}_\kappa$ suitable to characterize the motion. The fault detection estimates the level of unintended motion in the captured signal and hence provides an estimate $\hat{\xi}_\kappa$ for the symptom level.

where $\tilde{g}_f$ is a smooth mapping, which defines the temporal evolution of the internal motor state estimate $\hat{\xi}_{\kappa-1}$. Since the dynamics (2.12) only depends on the most recent previous internal motor state ξ_κ, it fulfills the Markov property. This property inherits to the fault detection,

$$\hat{\xi}_\kappa = \mathbb{E}[g_f(\hat{\xi}_{\kappa-1}, \mathbf{z}_\kappa)], \tag{2.16}$$

where $\hat{\xi}_{\kappa-1}$ contributes to the model-based prediction step and $\mathbf{z}_\kappa$ to the measurement based innovation. We call the subsystem consisting of system monitoring unit and fault detection unit a *symptom level estimator*. A schematic visualization of the estimator is provided in Figure 2.4.

Remark 2.3.2 *It is not possible to employ a state observer instead of the fault detection unit to estimate the symptom severity level ξ_κ in the dynamics (2.12), as a state observer controls its system model to track the unknown real system. In our system, however, the desired trajectory input as well as the external factors are unknown. Therefore, we can hardly reconstruct the unknown system.*

We propose to use GP modeling for the fault detection, i.e. to estimate the function g_f in equation (2.16). More details on the feature generation in the monitoring system and on how we employ GPs to achieve a reliable fault detection will be provided in Chapter 5.

2.4 Discussion

We introduced a holistic dynamics model for the human sensorimotor system in this section consisting of three subsystems, namely the motor execution dynamics, the decision process dynamics and the internal motor state dynamics. Our model plant is motivated by neurophysiological studies and verified findings on the human sensorimotor system. It covers all relevant aspects of the human sensorimotor system for describing the human motor and action behavior. Additionally, we provided approaches to approximate each of the three subsystems in the human plant model. Since, the subsystem concerning the human internal motor state is not accessible with non-invasive methods, we provided an estimator for this subsystem consisting of a monitoring unit and a fault detection. In the following Chapters 3, 4 and 5 we explicitly introduce the methodology for the model approximations, and we experimentally validate and evaluate each of the subsystems.

Alternatively, a plant model for the sensorimotor system could be generated employing traditional methods such as optimal feedback control, impedance control and predictive control. In complex and general task settings, however, these approaches did not achieve convincing results (see Section 1.2). Data-driven models, in contrast, are regarded suitable for modeling the human sensorimotor system. We are especially convinced by the approximation capabilities of GP models in nonlinear systems. Therefore, we rely on various forms of GP modeling in each of the subsystems $1-3$ of the human dynamics.

3

Gaussian Process for Rigid Motion Dynamics on $SE(3)$

In this chapter, we introduce the GP generalization for modeling 6-DoF rigid motions on the manifold $SE(3)$ after having reviewed related work on GPs and GP generalizations. We provide a formal GP definition and highlight characteristic GP model properties. Thereafter, we recapitulate GP model learning and prediction to point out the requirements for the aspired GP generalization. Before we introduce a set of fundamental functions containing the zero mean and a dot-product kernel as well as a stationary kernel function, we present two well suited rigid motion parametrizations, namely the axis-angle and Euclidean vector, and the dual quaternions. Next, we compare the presented GP generalization variants to assess their optimal application scope. Last, we introduce various integration methods from the Euclidean velocity space $TSE(3)$ to the state space $SE(3)$. We explain how the presented integrations are used to define Gaussian-like probability distributions on the rigid motions and detail the properties of different distributions. The integration methods complete the GP framework for modeling rigid motion dynamics, as they define pullback operators into the original space $SE(3)$ of the dynamics (2.4).

3.1 Related Work and Open Problems

The GP model was first introduced in 1996 by [63]. The authors Williams and Rasmussen suggested to replace supervised neural networks by GPs, as neural networks converge to GPs for increasing number of hidden layers in the model. Neal [64] showed in his work on Bayesian learning that the prior distribution over nonlinear functions implied by the Bayesian neural network is part of the GP probability distributions. Gibbs [65] was the first to introduce a kernel function, which constitutes the essential part of GP modeling, with spatially varying length scales. This innovation paved the way for the construction of new kernel functions from basic kernel functions as presented in [66, Chap. 6.2].

Even though since then, numerous publications on GP models flooded in, the traditional formulation of the GP in Euclidean vector space was rarely questioned. Hence, a major drawback of this model is that it is only defined in Euclidean space, even though a formulation with non-Euclidean input space is possible in principle.

Independently of the machine learning community developing the GP, the control community investigated a similar concept for supervised learning, namely system identification.

In the following, we disambiguate the system identification from the GP and provide an overview of possible generalization approaches of the GP model. We especially focus herein on kernel concepts on non-Euclidean space, as the kernel functions is essential in GP modeling.

Application of GPs to Dynamical Models versus System Identification

Recently, the GP, which is a Bayesian model, was shown to be suitable for modeling human motion [36]. Gaussian process dynamical models (GPDM) are used for motion estimation for instance in [36, 67]. These approaches employ linear dimensionality reduction methods in the GP regression for improved prediction results. As the lower dimensional spaces are assumed to be Euclidean, their suitability for modeling human behavior (i.e. motion on a non-Euclidean manifold) is limited. In parallel to the GP regression approach the control community developed system identification (SID) for inference of the system parameters. Also SID is a tool to derive dynamics models from data. Basically, it infers a model by data analysis [68], i.e. by assigning a best model out of a pre-selected set of models. Recently, limitations of the classical approach to linear and nonlinear system identification are described: A formulation of the identification problem in a fully Bayesian context is still lacking, see [69] and machine learning techniques are proposed as a promising alternative. Since its first application to dynamical systems in [60], the GP gains popularity in this domain.

Kernels on Manifolds

The kernel functions are considered as the core of GP modeling. However, kernels are nontrivial to extend beyond the Euclidean space in general. In [70], the subclass of characteristic kernels are investigated, i.e. where the mapping is injective. An example for a characteristic kernel is given on the special orthogonal group $SO(3)$ which represents rotations in $\mathbb{R}^3$. However, the correlation of orientations, described by the characteristic kernel, is not related to a distance function based on the rotation magnitude between orientations. The manifold metrics in a Hilbert space embedding, investigated in [71], do not necessarily result in valid kernel functions. One counter example is a kernel resulting from the Stein-divergence. A theorem that states sufficient and necessary conditions to obtain valid Gaussian kernels on metric spaces is introduced in [72]. We make use of their results in the development of our kernel function. The kernels introduced in [73], used for computing a nonparametric embedding of distributions do not require a Gaussian assumption. None of the kernel approaches listed so far was applied to GP models to the best of our knowledge.

Another interesting line of research considers manifold-based learning methods. The approaches [74–76] employ nonlinear dimensionality reduction methods and find application in machine learning as non-Gaussian manifold kernel methods. In contrast to these approaches, the manifold kernel dimension reduction [77] is a supervised setting for dimensionality reduction. The authors combine the approach with manifold regression through Laplacian eigenmaps. The kernel formulation of those methods admits an interpretation of the algorithms as warping of the input space into a feature space where the manifold is flat [78].

Gaussian Process Generalizations

As our interest lies on GP modeling, we survey approaches that incorporate GPs over non-Euclidean spaces in the following. So far, only a few works consider the problem of generalizing the GP model to non-Euclidean input space, even though it is crucial for achieving accurate estimation results. An embedding of latent variables into non-Euclidean space is presented for GP latent variable models in [79]. However, as only locally linear embeddings are considered, the standard GP structure is maintained. Another approach [80] introduces a geodesic GP. The GP model is used to train a surface embedding in Euclidean space. That is, the GP is defined in Euclidean space but maps geodesic distances to a Euclidean equivalent. Further, [81] aims to model complex and non-differentiable functions, for which the smoothness assumptions on standard kernel functions are unsuitable. A manifold GP is introduced that jointly models two composed functions $m \circ g$, where m is a deterministic mapping from an abstract manifold M into a Euclidean feature space and g the GP regression task. In contrast to this approach where both mappings m and g are abstract, we propose an alternative approach in which a concrete manifold is given. As we focus our investigations to generalized GP models over rigid motions, this allows us to describe the manifold M explicitly and to define an explicit mapping from the generalized GP output space to the manifold.

3.2 Gaussian Process Definition and Characteristics

Compactly stated, *a GP is a collection of random variables, of which any finite number have a joint Gaussian distribution* [43] and a precise, more explicit formulation is given in the following.

Definition 3.2.1 *Let $\mathcal{X}$ be a (multidimensional) index set, and denote by $\{\varphi(\mathbf{x})\}_{\mathbf{x}\in\mathcal{X}}$ a real-valued stochastic process over $\mathcal{X}$. Such a process is called Gaussian, if and only if any finite collection of random variables $\{\varphi(\mathbf{x}_1), \ldots, \varphi(\mathbf{x}_\nu)\}$ is ν-dimensional multivariate Gaussian distributed.*

A GP is fully specified by a mean function $m(\mathbf{x})$ and a kernel function $k(\mathbf{x}, \mathbf{x}')$ in function space [43]

$$\begin{aligned} m(\mathbf{x}) &= \mathbb{E}\left[\varphi(\mathbf{x})\right], \\ k(\mathbf{x}, \mathbf{x}') &= \mathbb{E}\left[(\varphi(\mathbf{x}) - m(\mathbf{x}))\,(\varphi(\mathbf{x}') - m(\mathbf{x}')\right], \end{aligned} \tag{3.1}$$

and thus, we can write

$$\varphi(\mathbf{x}) \sim \mathrm{GP}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')). \tag{3.2}$$

To understand the basic functioning of the GP, we have to move our point of view from function to feature space. The GP model essentially circumvents the intrinsic limitations in expressiveness of linear modeling by first projecting the inputs $\mathbf{x}, \mathbf{x}'$ into some (high) dimensional feature space using a kernel function $k(\cdot, \cdot)$ before employing a linear model there. The so called “kernel trick” allows for computationally efficient implicit calculations in the feature space [43]. Therefore, the essential part in GP modeling concerns the kernel $k(\cdot, \cdot)$. The mean function $m(\cdot)$ is often set to zero in practice, as this simplifies calculation without limiting the expressiveness of the process [66, Chap. 6.4.1].

Apart from few exceptions, the kernel functions can be divided in two classes: *stationary and dot-product kernels*. A kernel is called stationary, if the obtained covariances are invariant to translations in the input space. A dot-product kernel depends on the non-stationary inner product between inputs. Typically, non-constant stationary kernels depend on some kind of distance measure between input samples. Part of the most widely used stationary kernel functions in machine learning is the squared exponential kernel

$$k_{\mathrm{eu}}(\mathbf{x},\, \mathbf{x}') = \sigma_f^2\, \exp\left(-\frac{[d_{\mathrm{eu}}(\mathbf{x},\, \mathbf{x}')]^2}{2\lambda^2}\right) + \sigma_n^2 \delta_{ij}, \tag{3.3}$$

where $d_{\mathrm{eu}}(\mathbf{x},\, \mathbf{x}') = \|\, \mathbf{x} - \mathbf{x}'\|$ is the Euclidean distance and $\mathbf{h} = (\lambda, \sigma_f, \sigma_n)$ the hyperparameter vector consisting of length-scale λ, signal variance σ_f and signal noise σ_n.

We use the standard formulation of introducing a noise term σ_n in the kernel function as required for (2.4), see [43]. Specifically, the δ-function is defined by

$$\delta_{ij} = \begin{cases} 1, & \text{if } i = j, \\ 0, & \text{if } i \neq j. \end{cases} \tag{3.4}$$

To provide an intuitive understanding of the principle of GP modeling, we detail the steps model learning and prediction in the following. Then, we explain the traditional way of modeling rigid motions using a GP.

3.2.1 Model training

Model training ultimately refers to optimizing the parameters of the mean and the kernel function. We consider a smooth but unknown mapping $f : \mathcal{X} \to \mathcal{V}$, with $f(\mathbf{x}) = \dot{\mathbf{x}}$. Let $\mathcal{D}_\nu = \{(\mathbf{x}_i,\ \dot{\mathbf{x}}_i)\}_{i=1}^{\nu}$ be a training data set consisting of n-dimensional input-output pairs, where the output measurements possibly are noisy. In the vector-valued stochastic processes $f = (\varphi_1, \ldots, \varphi_n)^\top$ each component φ_l is approximated by a separate Gaussian process $\varphi_l(\mathbf{x}) \sim \mathrm{GP}(m_l(\mathbf{x}),\, k_l(\mathbf{x},\, \mathbf{x}'))$ employing the same class of mean functions $m_l(\cdot)$ and kernel functions $k_l(\cdot, \cdot)$ for all $l = 1, \ldots, n$. Per output dimension l, a set of GP hyperparameters $\mathbf{h}_l$, configuring the pre-identified classes of mean and kernel functions, is adjusted to approximate the dynamics f. Optimal approximation of f, given the input values $\mathbf{x}_i$, is obtained for the dimension-wise globally maximum likelihoods of the observations $\dot{\mathbf{x}}_i$. Usually, the hyperparameter optimization is performed using standard gradient decent methods to minimize the negative marginal log likelihood, which can be calculated analytically,

$$\mathbf{h}_l^* = \underset{\mathbf{h}_l}{\operatorname{argmin}} - \log p\left(\{(\dot{\mathbf{x}}_i)_l\}_{i=1}^{\nu}\,\middle|\,\{\mathbf{x}_i\}_{i=1}^{\nu}, \mathbf{h}_l\right). \tag{3.5}$$

Remark 3.2.1 *In case we would like use another Bayesian method to model the mapping for the training data set $\mathcal{D}_\nu = \{(\mathbf{x}_i,\ \dot{\mathbf{x}}_i)\}_{i=1}^{\nu}$ we could employ for instance latent variable models, i.e. a model that incorporates variables which are not directly observable, or probabilistic clustering, i.e. grouping data into a finite set of categories based on a probabilistic rule (more details see Section 4.2.1), or GMMs, i.e. a finite convex combination of (multivariate) Gaussian distributions. The most prominent alternative certainly is the GMM. To train the model, one employs the Expectation-Maximization (EM) algorithm which iteratively calculates the* *probability of each data point to belong to a particular category, and optimizes the parameters of the Gaussian distribution to best describe the weighted data points. In contrast to the GP, which provides a probability distribution over outputs given the inputs,* $\mathbf{p}\left(\{\dot{\mathbf{x}}_i\}_{i=1}^{\nu}\middle|\{\mathbf{x}_i\}_{i=1}^{\nu}\right)$*, the GMM learns the probability over the whole data set,* $\mathbf{p}\left(\{\mathbf{x}_i,\, \dot{\mathbf{x}}_i\}_{i=1}^{\nu}\right)$*.*

3.2.2 Prediction

For any new input value $\mathbf{x}_{\nu+1} \notin \mathcal{D}_\nu$ the GP returns a predictive Gaussian probability distribution defined in the target set $\mathcal{V}_l$ for each component $l \in \{1,\, \dots,\, n\}$ of the n-dimensional output value $\dot{\mathbf{x}}_{\nu+1}$, namely $\mathcal{N}(\mu_l,\, \sigma_l)$. Mean μ_l and variance σ_l of the l-th component are calculated analytically by

$$\begin{aligned}\mu_l &= m_l(\mathbf{x}_{\nu+1}) + \mathbf{K}_*\mathbf{K}^{-1}\begin{pmatrix}(\dot{\mathbf{x}}_1)_l - m_l(\mathbf{x}_1)\\ \vdots \\ (\dot{\mathbf{x}}_\nu)_l - m_l(\mathbf{x}_\nu)\end{pmatrix},\\ \sigma_l &= k_l(\mathbf{x}_{\nu+1},\, \mathbf{x}_{\nu+1}) - \mathbf{K}_*\mathbf{K}^{-1}\mathbf{K}_*^\top,\end{aligned} \tag{3.6}$$

where the Gram matrix $\mathbf{K}$ is obtained by $(\mathbf{K})_{ij} = k_l(\mathbf{x}_i,\, \mathbf{x}_j)$ for $i, j \in \{1, \dots, \nu\}$ and each entry of the row vector $(\mathbf{K}_*)_j = k_l(\mathbf{x}_{\nu+1},\, \mathbf{x}_j)$. These component-wise estimates in form of scalar predictive Gaussian probability distributions are joint into one multivariate Gaussian distribution

$$\dot{\mathbf{x}}_{\nu+1} \sim \mathcal{N}(\mu,\, \Sigma), \tag{3.7}$$

where $\mu := (\mu_1, \dots, \mu_n)^\top$ and σ_l are the diagonal entries of the otherwise empty covariance matrix Σ.

Remark 3.2.2 *Prediction in the GMM alternative: The training data representation in the learned GMM implies the prediction output to generate completely unseen input-output pairs* $(\mathbf{x}_{\nu+1},\, \dot{\mathbf{x}}_{\nu+1})$ *by random sampling from the GMM distribution. Hence, we cannot fix the input* $\mathbf{x}_{\nu+1}$ *of the unknown function* $\check{f}$*.*

3.2.3 Classic 6-DoF Gaussian Process

In real world applications it is common procedure to encode rigid motions in a 6D state vector $\mathfrak{g} = (\phi,\, \psi,\, \vartheta,\, v_x,\, v_y,\, v_z)^\top$. Hence, the state vector $\mathfrak{g}$ is treated inaccurately as if it would be defined in the Euclidean space $\mathbb{R}^6$. Employing (3.2) induces the classic GP for 6-DoF rigid motions

$$f(\mathfrak{g}) = \mathrm{GP}_{\mathbb{R}^6}(m(\mathfrak{g}), k(\mathfrak{g},\, \mathfrak{g}')). \tag{3.8}$$

To distinguish the classic 6-DoF GP from the generalized GPs defined on $SE(3)$, which we will introduce later on, we denote the classic one by $\mathrm{GP}_{\mathbb{R}^6}$.

3.3 Representations of $SE(3)$

We discuss various representations of $\mathfrak{g}$ in $SE(3)$ in this section to obtain a more concrete description of the dynamical system (2.4). Thereby, we attach importance to certain properties of the parametrizations. We desire an efficient combination of rotation and translation into a unified structure $\mathfrak{g} \in SE(3)$, and we require a dot product $\langle \mathfrak{g}, \mathfrak{g}' \rangle$ and a metric $d(\mathfrak{g}, \mathfrak{g}')$ to be defined. Additionally, we expect the representation to include a computationally efficient composition of rigid motions and integration from velocity space $TSE(3)$ to the special Euclidean group $SE(3)$. Following, we eliminate representations with major deficiencies and nominate the ones that best meet the requested properties.

The most popular pose representation for $\mathfrak{g} \in SE(3)$ is a vector of Euler angles concatenated with translation vector $\mathbf{t}$, i.e. $\mathfrak{g} = (\phi, \psi, \vartheta, v_x, v_y, v_z)^\top$. However, the rotation angles are not Euclidean by definition and thus, the Euclidean distance between Euler angles is erroneous. Therefore, we exclude this representation from further investigations.

Another possible alternative is the representation of $SE(3)$ in terms of a 4×4 homogeneous transformation matrix $\mathfrak{g} = \left(\begin{smallmatrix} \mathbf{R} & \mathbf{v} \\ 0 & 1 \end{smallmatrix}\right)$, where $\mathbf{R}$ a rotation matrix and $\mathbf{v} \in \mathbb{R}^3$ a translation vector. Using this parametrization, the tangent space representation is obtained by left transition $\mathfrak{g}$ of the Lie algebra, $T_{\mathfrak{g}}SE(3) = \mathfrak{g}\,\mathfrak{se}(3)$, where the Lie algebra is represented by the set of twists,

$$\mathfrak{se}(3) = \left\{ \begin{pmatrix} \Omega & \boldsymbol{\tau} \\ 0 & 0 \end{pmatrix} \middle| \Omega \in \mathbb{R}^{3\times 3}, \Omega^\top = -\Omega, \boldsymbol{\tau} \in \mathbb{R}^3 \right\}. \tag{3.9}$$

The vector $\boldsymbol{\tau} = (\tau_x, \tau_y, \tau_z)^\top$ represents the linear velocities, and the skew symmetric matrix Ω encodes the angular velocities ω_ϕ, ω_ψ, ω_ϑ via

$$\Omega = \begin{pmatrix} 0 & \omega_\phi & \omega_\psi \\ -\omega_\phi & 0 & \omega_\vartheta \\ -\omega_\psi & -\omega_\vartheta & 0 \end{pmatrix}. \tag{3.10}$$

A use of this representation is theoretically plausible, but involves handling bulky 4×4 matrices, non-intuitive distance measures and cumbersome matrix integration. Instead of deepening the investigations using homogeneous transformation matrices, we focus on two further rigid motion parametrizations.

The axis-angle orientation representation is canonically combined with a translation vector. This representation is minimal, i.e. the rotation and translation representation has each 3 free parameters only. Further, it allows for a particularly efficient and distance-preserving integration from velocity space to the space of rigid motions. On the downside, no efficient composition algorithm is available, as the vector sum is not employable. Alternatively, there is the dual quaternion representation $\mathfrak{g} \in \mathbb{H}_D$ for 6-DoF rigid body motions, which are derived naturally from unit quaternions representing rotations. Unit quaternions produce smooth rotations and can be efficiently composed via quaternion multiplication. They extend naturally to dual quaternions, while preserving their properties. Hence, they permit encapsulating rotation and translation in a unified representation [82] in a close to minimal representation with 7 free parameters. However, the integration from velocity space into the manifold of rigid motions $SE(3)$ employing a global canonical projection is no isometry, i.e. it is not distance-preserving.

In the following, we present the dual quaternion and axis-angle and translation vector representation. For each of the parametrizations we highlight the linkage of rotation and translation and introduce a rigid motion composition and a dot product. The metrics on the rigid motions for both representations will be introduced in the subsequent Section 3.4.2 and the integration will be dealt with extensively in Section 3.5, where we complete the framework by defining a pullback into the original space $SE(3)$ of the dynamics (2.4).

3.3.1 Dual Quaternions

We briefly recapitulate two special types of quaternions $\mathbb{H}$, the unit quaternions and the pure imaginary quaternions. Those two special cases together joint into one structure define the dual quaternions, a unified manner of representing rigid motions $\mathfrak{g} \in SE(3)$.

The quaternions $\mathbb{H}$ define a skew field, i.e. a ring in which division is defined. Hence they fulfill more properties than standard rings (e.g. the matrix ring), but they lack the multiplication commutativity of fields. The skew field of quaternions is defined by

$$\mathbb{H} := \left\{ \mathbf{q} \;\middle|\; \mathbf{q} = q_w + q_x i_x + q_y i_y + q_z i_z \;\wedge\; q_w,\, q_x,\, q_y,\, q_z \in \mathbb{R} \right\}. \tag{3.11}$$

It extends the field of complex numbers to three imaginary dimensions at the expense of its commutativity. Since required for the generalized GP kernel definition, we additionally present the dot product between quaternions,

$$\langle\, \mathbf{q}_1,\, \mathbf{q}_2 \rangle = q_{w_1} q_{w_2} + q_{x_1} q_{x_2} + q_{y_1} q_{y_2} + q_{z_1} q_{z_2}. \tag{3.12}$$

Besides, the special case of (3.12), where twice the same quaternion is employed, defines the quaternion norm $\|\, \mathbf{q}\| = \sqrt{\langle\, \mathbf{q},\, \mathbf{q} \rangle} = \sqrt{\mathbf{q}\,\overline{\mathbf{q}}}$, where $\overline{\mathbf{q}} = q_{w_2} - q_{x_2} i_x - q_{y_2} i_y - q_{z_2} i_z$ is the quaternion conjugate of $\mathbf{q}$.

A unit quaternion $\mathbf{q}_{\mathrm{rot}} \in \mathbb{H}$ is a quaternion that additionally satisfies the property $\|\, \mathbf{q}_{\mathrm{rot}}\| = 1$. Hence, the set of unit quaternions conforms to the unit hypersphere S_3. A strong relation exists between the unit quaternions S_3 and the special orthogonal group $SO(3)$, the group of rotations: The unit quaternions S_3 are a double coverage of the group $SO(3)$. This means, any unit quaternion $\mathbf{q}_{\mathrm{rot}} \in S_3$ uniquely represents a 3D rotation in $SO(3)$, but the representation of rotations is not uniquely defined by unit quaternions. The group structure of $SO(3)$ is obtained from the operation composition of rotations $\circ$, which is simply inherited by the unit quaternion through quaternion multiplication $\mathbf{q}_{\mathrm{rot}_1}\, \mathbf{q}_{\mathrm{rot}_2}$, i.e.

$$\begin{aligned} \mathbf{q}_{\mathrm{rot}_1} \circ \mathbf{q}_{\mathrm{rot}_2} = {} & q_{w_1} q_{w_2} - q_{x_1} q_{x_2} - q_{y_1} q_{y_2} - q_{z_1} q_{z_2} \\ & + (q_{w_1} q_{x_2} + q_{x_1} q_{w_2} + q_{y_1} q_{z_2} - q_{z_1} q_{y_2}) i_x \\ & + (q_{w_1} q_{y_2} - q_{x_1} q_{z_2} + q_{y_1} q_{w_2} + q_{z_1} q_{x_2}) i_y \\ & + (q_{w_1} q_{z_2} + q_{x_1} q_{y_2} - q_{y_1} q_{x_2} + q_{z_1} q_{w_2}) i_z. \end{aligned} \tag{3.13}$$

An imaginary quaternion $\mathbf{q}_{\mathrm{tra}} \in \mathbb{H}$ is a quaternion, where the real parameter $q_w = 0$, i.e. $\mathbf{q}_{\mathrm{tra}} = q_x i_x + q_y i_y + q_z i_z$. The imaginary quaternions are isomorphic to the space of translations $\mathbb{R}^3$. Thus, a translation vector $\mathbf{v} = (v_x,\, v_y,\, v_z)^\top \in \mathbb{R}^3$ is uniquely represented by an imaginary quaternion $\mathbf{q}_{\mathrm{tra}} = v_x i_x + v_y i_y + v_z i_z$ and vice versa. The composition of imaginary quaternions is defined by quaternion addition, that is component wise addition in the vector space to the basis $\{i_x, i_y, i_z\}$

$$\mathbf{q}_{\mathrm{tra}_1} \circ \mathbf{q}_{\mathrm{tra}_2} = (q_{x_1} + q_{x_2}) i_x + (q_{y_1} + q_{y_2}) i_y + (q_{z_1} + q_{z_2}) i_z \tag{3.14}$$

and the imaginary quaternion dot product likewise reduces to a 3D vector dot product

$$\langle\, \mathbf{q}_{\mathrm{tra}_1},\, \mathbf{q}_{\mathrm{tra}_2} \rangle = q_{x_1} q_{x_2} + q_{y_1} q_{y_2} + q_{z_1} q_{z_2}. \tag{3.15}$$

We are interested in representing rotation and translation in a unified manner. Therefore, we investigate the defining spaces of the representations and the rigid motions $SE(3)$. Together, rotation and translation define a rigid motion. Thus, the representations

$$\{(\, \mathbf{q}_{\mathrm{rot}},\, \mathbf{q}_{\mathrm{tra}})\} \simeq S_3 \times \mathbb{R}^3 \tag{3.16}$$

are homeomorphic. Further, the special Euclidean group $SE(3)$ is homeomorphic to $SO(3) \times \mathbb{R}^3$, where $\times$ denotes the direct Cartesian set product. As the unit quaternions S_3, however, cover the rotations $SO(3)$ twice, we need to divide the sphere S_3 into the equivalence classes $\{\pm 1\}$ to obtain an homeomorphism $S_3/\{\pm 1\} \simeq SO(3)$. Using this trick, we loose the direct product as link between the rotation and translation space, due to the ambiguity in the unit quaternion rotation representation.

Proposition 3.3.1 *The dual quaternions representing rigid motions are homeomorphic to the special Euclidean group, if we employ the semi-direct product $\rtimes$ to connect the spaces S_3 and $\mathbb{R}^3$,*

$$SE(3) \simeq S_3/\{\pm 1\} \rtimes \mathbb{R}^3. \tag{3.17}$$

Thus, any rigid motion $\mathfrak{g} \in SE(3)$ can be equivalently represented by $\mathfrak{g} \equiv (\, \mathbf{q}_{rot},\, \mathbf{q}_{tra}) \in S_3/\{\pm 1\} \rtimes \mathbb{R}^3$.

To represent the tuple $(\, \mathbf{q}_{\mathrm{rot}},\, \mathbf{q}_{\mathrm{tra}})$ in a unified structure, we employ a dual extension of the quaternions $\mathbb{H}$, namely the ring of the dual quaternions $\mathbb{H}_D$, which is defined by

$$\mathbb{H}_D = \{\, \mathfrak{g} \mid\, \mathfrak{g} = \mathbf{q}_{\mathrm{re}} + \epsilon\, \mathbf{q}_{\mathrm{du}} \;\wedge\; \mathbf{q}_{\mathrm{re}},\, \mathbf{q}_{\mathrm{du}} \in \mathbb{H}\}, \tag{3.18}$$

where $\mathbf{q}_{\mathrm{re}}$ and $\mathbf{q}_{\mathrm{du}}$ denote quaternions in the real and the dual space, respectively, and ϵ is a dual unit which holds $\epsilon^2 = 0$ (for more details see [83]). The dual quaternion $\mathfrak{g}$ representing the rigid motion $(\, \mathbf{q}_{\mathrm{rot}},\, \mathbf{q}_{\mathrm{tra}})$, is composed by

$$\mathfrak{g} := \underbrace{\mathbf{q}_{\mathrm{rot}}}_{\mathbf{q}_{\mathrm{re}}} + \epsilon \underbrace{\frac{1}{2}\, \mathbf{q}_{\mathrm{tra}}\, \mathbf{q}_{\mathrm{rot}}}_{\mathbf{q}_{\mathrm{du}}}. \tag{3.19}$$

The composition of dual quaternions, denoting successive execution of rigid motions, is defined by dual quaternion multiplication of $\mathfrak{g}_1$ and $\mathfrak{g}_2$, calculated as

$$\mathfrak{g}_1 \circ \mathfrak{g}_2 = \mathbf{q}_{\mathrm{re}_1} \mathbf{q}_{\mathrm{re}_2} + \epsilon\, (\, \mathbf{q}_{\mathrm{re}_1} \mathbf{q}_{\mathrm{du}_2} + \mathbf{q}_{\mathrm{du}_1} \mathbf{q}_{\mathrm{re}_2}). \tag{3.20}$$

Additionally, the dual quaternion conjugate of $\mathfrak{g}$ is calculated as $\overline{\mathfrak{g}} = \overline{\mathbf{q}}_{\mathrm{re}} + \epsilon\, \overline{\mathbf{q}}_{\mathrm{du}}$, and finally, the dual quaternion dot product is given analog to the dot product in vector spaces

$$\langle\, \mathfrak{g}_1,\, \mathfrak{g}_2 \rangle = \langle\, \mathbf{q}_{\mathrm{re}_1},\, \mathbf{q}_{\mathrm{re}_2} \rangle + \langle\, \mathbf{q}_{\mathrm{du}_1},\, \mathbf{q}_{\mathrm{du}_2} \rangle, \tag{3.21}$$

where both addends on the right define quaternion scalar products according to (3.12).

3.3.2 Axis-angle and Translation Vectors

From Euler's fixed point theorem (1776) [84] we know that in 3D space any rigid-body displacement with one fixed point, i.e. any composition of rotations, can be equivalently described by a single rotation about some non-trivial axis of rotation through the fixed point. This axis is also called Euler axis. Thus, the set of unit length Euler axes $\mathbf{u}$ together with a rotation angle θ parametrizes the rotation group, called special Orthogonal group $SO(3)$

$$SO(3) \subset \left\{ \theta \mathbf{u} \in \mathbb{R}^3 \;\middle|\; \|\mathbf{u}\| = 1 \wedge \theta \in [0, \pi] \right\}. \tag{3.22}$$

The set given in (3.22) defines the solid ball $B_\pi(0)$ in $\mathbb{R}^3$ with radius $0 \leq r \leq \pi$ centered around the origin and is thus closed, dense and compact. Ambiguity in the representation occurs for $\theta = \pi$, as $\pi\mathbf{u} = -\pi\mathbf{u}$ define the same rotation. To obtain an homeomorphism between the rotation group $SO(3)$ and the axis-angle representation, we additionally fix the Euler axis representation for $\theta = \pi$ and obtain

$$\begin{aligned} \tilde{B}_\pi(0) := B_\pi(0) \setminus \Big\{ \pi\mathbf{u} \;\Big|\; & \mathbf{u}_z < 0 \vee (\mathbf{u}_z = 0 \wedge \mathbf{u}_y < 0) \\ & \vee (\mathbf{u}_z = \mathbf{u}_y = 0 \wedge \mathbf{u}_x < 0) \Big\}. \end{aligned} \tag{3.23}$$

This parametrization of the rotation group by a Euler axis and a rotation angle is a minimal and unique, $SO(3) \simeq \tilde{B}_\pi(0)$.

The special orthogonal group $SO(3)$ obtains its group structure from the operation composition of rotations. As the axis-angle representation only defines a pseudo-vector, none of the vector operations addition and multiplication is suitable for rotation composition. Instead, we define the rotation composition of two rotations $\theta_1\mathbf{u}_1$ and $\theta_2\mathbf{u}_2$ in $\tilde{B}_\pi(0)$, inspired by unit quaternion multiplication,

$$\theta_2\mathbf{u}_2 \circ \theta_1\mathbf{u}_1 = \left(\underbrace{2\,\mathrm{acos}(\left|a - bc\right|)}_{\theta_3} \, \underbrace{\frac{\sqrt{1-a^2} + b(\mathbf{u}_2 \times \mathbf{u}_1)}{\sqrt{1-(a^2-bc)^2}}}_{\mathbf{u}_3} \right), \tag{3.24}$$

where we substitute $a = \cos\left(\frac{\theta_2+\theta_1}{2}\right)$, $b = \sin\left(\frac{\theta_2}{2}\right)\sin\left(\frac{\theta_1}{2}\right)$ and $c = 1 + \mathbf{u}_2\,\mathbf{u}_1$. The operation $\times$ defines the cross product of vectors $\mathbf{u}_2$ and $\mathbf{u}_1$.

We additionally introduce the dot product between rotations for the axis-angle representation, as we aspire to employ it in the GP generalization. The dot product of two rotations in axis-angle is defined as the regular scalar product in vector spaces,

$$\langle \theta_1\mathbf{u}_1, \theta_2\mathbf{u}_2 \rangle = |\theta_1|\,|\theta_2| \cos\sphericalangle\, \mathbf{u}_1\mathbf{u}_2. \tag{3.25}$$

In the following, we present our approach to combine rotation with translation, which we represent by Euclidean vectors $\mathbf{v} \in \mathbb{R}^3$.

Proposition 3.3.2 *As $\tilde{B}_\pi(0)$ is a unique rotation representation, the link with translations can be realized using the standard Cartesian set product. The spaces of rotation $\tilde{B}_\pi(0)$ and translation $\mathbb{R}^3$ jointly define a homeomorphism to the special Euclidean group*

$$SE(3) \simeq \tilde{B}_\pi(0) \times \mathbb{R}^3 \tag{3.26}$$

and any rigid motion $\mathfrak{g} \in SE(3)$ can be equivalently represented by $\mathfrak{g} \equiv (\theta\mathbf{u},\, \mathbf{v}) \in \tilde{B}_\pi(0) \times \mathbb{R}^3$.

For the axis-angle representation no dual extension is available to combine rotation and translation in a unified manner and the pair $(\theta\mathbf{u},\, \mathbf{v})$ is kept separate.

The group structure on $SE(3)$ is inherited from the group structure on the subspaces, namely composition. As translation composition in $\mathbb{R}^3$ is defined via vector addition, we obtain as composition operation of rigid motions in $SE(3)$

$$(\theta_3\mathbf{u}_3,\, \mathbf{v}_3) = \Big(\theta_2\mathbf{u}_2 \circ \theta_1\mathbf{u}_1, \theta_2\mathbf{u}_2(\, \mathbf{v}_1) +\, \mathbf{v}_2\Big), \tag{3.27}$$

where $\theta_2\mathbf{u}_2 \circ \theta_1\mathbf{u}_1$ as defined in (3.24) and $\theta_2\mathbf{u}_2(\, \mathbf{v}_1)$ is obtained using the Rodrigues rotation formula

$$\theta_2\mathbf{u}_2(\mathbf{v}_1) = \mathbf{v}_1 \cos\theta_2 + \sin\theta_2(\mathbf{u}_2 \times\, \mathbf{v}_1) +\, \mathbf{u}_2(\mathbf{u}_2^\top \mathbf{v}_1)(1 - \cos\theta_2) \tag{3.28}$$

for translation vectors $\mathbf{v}_i \in \mathbb{R}^3$, $i = 1, 2$. The dot product on $SE(3)$ with the representation by axis-angle pseudo-vector and translation vector is defined as in regular vector spaces,

$$\langle\, \mathfrak{g}_1,\, \mathfrak{g}_2\rangle = \langle\theta_1\mathbf{u}_1, \theta_2\mathbf{u}_2\rangle + \langle\, \mathbf{v}_1,\, \mathbf{v}_2\rangle. \tag{3.29}$$

Thus, the axis angle and translation vector representation defines a group on $SE(3)$ with the composition $\circ$ as group operation on the rigid motions.

Remark 3.3.1 *We introduced the most important rigid motion parametrizations: the Euler angles concatenated with translation vector, the homogeneous transformation matrix, the axis-angle pseudo vector concatenated with translation vector and the dual quaternions, a dual unification of a unit and an imaginary quaternion. The latter two meet our requirements to efficiently compose and integrate elements and to allow for a dot product* $\langle\, \mathfrak{g},\, \mathfrak{g}'\rangle$ *and a metric* $d(\, \mathfrak{g},\, \mathfrak{g}')$ *definition.*

3.4 Gaussian Process Generalization

In this section we introduce our approach to generalize GPs to modeling rigid motion dynamics. We first explain the fundamental basis of the generalization and understand that it is sufficient to define valid mean functions and kernel functions on the manifold to obtain the generalization to 6-DoF motion dynamics. Thereafter, we detail the steps of the procedure by presenting distance metrics on the manifold required for generalized stationary kernel functions. Finally, we introduce a set of fundamental mean and kernel functions containing the zero mean and a dot-product kernel as well as a stationary kernel function, from which more kernels can be constructed.

3.4.1 Mathematical Basis Concept

We start by investigating the role of the input space $\mathcal{X}$ in the GP model training. Recall, that the characteristic functions (mean and kernel), which we introduced in Section 3.2, solely depend on the GP domain $\mathcal{X}$. So far the vast majority of characteristic functions assumes domain $\mathcal{X}$ to equal $\mathbb{R}^n$. However, this is not a necessary requirement, which is evident from the definition of the characteristic functions (3.1): To obtain mean and kernel functions of a generalized GP, it suffices to define the functions in the new non-Euclidean GP domain $\mathcal{X}$, i.e. $m : \mathcal{X} \to \mathbb{R}$ and $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}_0^+$. Additionally, to assure the characteristic functions to be valid, the characteristic properties need to be fulfilled, which means the kernel is required to be symmetric and positive semi-definite [43]. Consequently, it suffices to provide valid mean functions and kernel functions defined on a domain $\mathcal{X}$ to generalize the GP model training to that domain.

Next, we analyze the model prediction in the output space $\mathcal{V}$. The necessary and sufficient requirement for the output space $\mathcal{V}$ is to allow the definition of a predictive Gaussian distribution, see (3.6), which in turn requires a real underlying vector space structure to be correctly normalized. As we selected the rigid motion dynamics (2.5) to be defined in continuous space the co-domain $TSE(3)$ provides the required structure. Thus, the requirement is fulfilled and the predictive output is defined.

Hence, it is sufficient to define valid mean functions and kernel functions to obtain the generalization to 6-DoF motion dynamics.

3.4.2 Metrics and Distance Measures

In this section, we investigate metrics and distance measures on the in GP input space $SE(3)$, as the prominent squared exponential kernel and most other stationary kernel functions depend on some kind of distance between input samples. We start by recapitulating the definition of a metric. Then, we introduce metrics over pure rotations before we extend them to full rigid motions. First the unit and dual quaternion representations are used, second the axis-angle rotation representation and its combination with translation is employed.

Metric Definition

A metric is a function, which defines a distance d between each pair of elements in some set $\mathcal{S}$ (in our case the set is the special orthogonal group $SO(3)$ or the space of rigid motions $SE(3)$, respectively).

The function $d : \mathcal{S} \times \mathcal{S} \to \mathbb{R}_0^+$ defines a metric, if for all elements $\mathbf{x}, \mathbf{y}, \mathbf{z} \in \mathcal{S}$ the following conditions are satisfied:

$$\begin{aligned} \text{non-negativity} \quad & d(\mathbf{x}, \mathbf{y}) \geq 0 \text{ and } d(\mathbf{x}, \mathbf{y}) = 0 \text{ iff } \mathbf{x} = \mathbf{y} \\ \text{symmetry} \quad & d(\mathbf{x}, \mathbf{y}) = d(\mathbf{y}, \mathbf{x}) \\ \text{triangle inequality} \quad & d(\mathbf{x}, \mathbf{z}) \geq d(\mathbf{x}, \mathbf{y}) + d(\mathbf{y}, \mathbf{z}). \end{aligned}$$

Note, the lower two conditions together define positive definiteness of the function d. This will be an important property later on for kernel functions.

Unit Quaternion Metrics

We contrast three distance measures on the hypersphere S_3. For the sake of clarity, the index "rot" for quaternions $\mathbf{q}_{\text{rot}} \in S_3$ is omitted in this section.

Figure 3.1 visualizes three distance measures that define metrics on the unit sphere S_3, i.e. they are non-negative, symmetric and fulfil the triangle inequality. All three metrics are equally applicable in a GP generalization, but differ essentially in the underlying concept. The quaternion norm

$$d_{\text{norm}}(\mathbf{q}, \mathbf{q}') = \sqrt{\mathbf{q}\,\overline{\mathbf{q}'}} = \|\mathbf{q} - \mathbf{q}'\|, \tag{3.30}$$

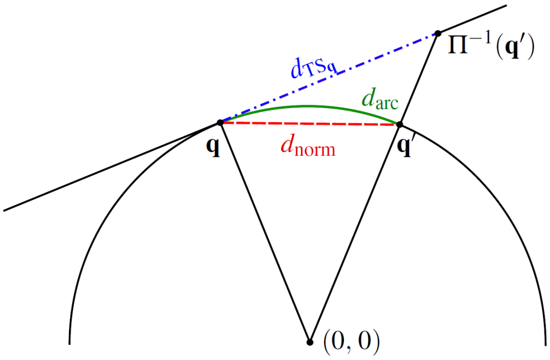


Figure 3.1: The distance measures consisting of quaternion norm, arc length and projection to the tangent space are depicted by a red dashed, green solid and blue dashed-dotted line, respectively.

where $\mathbf{q},\, \mathbf{q}' \in S_3$ (depicted by the red dashed line in Figure 3.1) provides identical results to the regular vector norm of points in the space $\mathbb{R}^4$. Its pros concern the low computational cost and the direct applicability of theoretical findings that are valid in Euclidean space. Contrary to intuition, however, the quaternion norm metric yields

$$2\, d_{\text{norm}}(\mathbf{q},\, \mathbf{q}_\Delta\, \mathbf{q}) > d_{\text{norm}}(\mathbf{q}, (\mathbf{q}_\Delta\, \mathbf{q}_\Delta)\, \mathbf{q}), \tag{3.31}$$

when applying twice a non-trivial rotation $\mathbf{q}_\Delta \neq (1,0,0,0)^\top$. The arc metric (depicted by the green solid line) between unit quaternions $\mathbf{q},\, \mathbf{q}' \in S_3$ is calculated via

$$d_{\text{arc}}(\mathbf{q},\, \mathbf{q}') = \min\, |\arccos(\pm\, \mathbf{q}_{\Delta w})|, \tag{3.32}$$

where $\mathbf{q}_{\Delta w}$ is the real part of the transforming unit quaternion $\mathbf{q}_\Delta = \mathbf{q}'\overline{\mathbf{q}}$ from $\mathbf{q}$ to $\mathbf{q}'$. It is an alternative metric, which satisfies equality in (3.31) for all $\mathbf{q},\, \mathbf{q}' \in \mathbb{H}$ with $\langle \mathbf{q},\, \mathbf{q}' \rangle \geq 0$. Both distance measures introduced so far are bounded, in fact $\max(d_{\text{norm}}) = \sqrt{2}$ and $\max(d_{\text{arc}}) = \frac{\pi}{2}$. An unbounded metric can be obtained when one quaternion, for instance $\mathbf{q}'$ is projected to the other's tangent space $TS_\mathbf{q}$. In Section 3.5.1 we will present the central projection Π, generating infinite distance for π-rotations. The origin of $TS_\mathbf{q}$ is identical to quaternion $\mathbf{q}$ and thus, the tangent space distance $d_{TS_\mathbf{q}}(\mathbf{q},\, \mathbf{q}')$ between projected quaternions is computed using the Euclidean norm in the space $TS_\mathbf{q} \simeq \mathbb{R}^3$,

$$d_{TS_\mathbf{q}}(\mathbf{q},\, \mathbf{q}') = \|\Pi^{-1}_\mathbf{q}(\mathbf{q}')\| = \|\, \mathbf{q}'_{TS_\mathbf{q}}\| = \|\Pi^{-1}_{\mathbf{q}'}(\mathbf{q})\|. \tag{3.33}$$

The appropriate choice of the metric is application dependent. While the tangent space norm $d_{TS_\mathbf{q}}$ assumes identical costs for a π-rotation and infinite translation, the quaternion norm d_{norm} produces only little inequality in (3.31) for small rotations. We suggest the use of the arc metric d_{arc} in all other applications.

Proposition 3.4.1 *The distance measures* (3.30),(3.32) *and* (3.33) *define metrics.*

Proof 3.4.1 (Proposition 3.4.1) *The proof for* (3.30) *and* (3.33) *to define metrics is trivial, as both distance measures are vector norms in the Euclidean space. The arc distance* (3.32) *is non-negative by definition of the* arccos *function, it is symmetric, as* $\mathbf{q}_\Delta^{-1} = \overline{\mathbf{q}}_\Delta$, *which does not affect the real part* $\mathbf{q}_{\Delta w}$ *of the quaternion, and a proof for the triangle inequality to be valid is provided in [85, pp. 195-196].* ☐

Dual Quaternion Distance Measure

Motivated by the fact that the magnitude of the transforming unit quaternion results in the topologically correct distance measure of rotations, we propose to base the dual quaternion distance on a magnitude measure between rigid motions $\mathfrak{g}$, $\mathfrak{g}' \in \mathbb{H}_D$,

$$\mathfrak{g}_\Delta = \overline{\mathfrak{g}} * \mathfrak{g}'. \tag{3.34}$$

The transforming dual quaternion $\mathfrak{g}_\Delta$ has to be applied to $\mathfrak{g}$, to arrive in $\mathfrak{g}'$. As the dual quaternion norm equals 1 for all rigid motions, we decompose $\mathfrak{g}_\Delta = \mathbf{q}_{\mathrm{re}_\Delta} + \epsilon\, \mathbf{q}_{\mathrm{du}_\Delta}$ into rotation $\mathbf{q}_{\mathrm{rot}_\Delta}$ and translation $\mathbf{q}_{\mathrm{tra}_\Delta}$ using (3.19) and define the transformation magnitude measure via

$$d_{\mathrm{mag}}(\mathfrak{g},\, \mathfrak{g}') = \sqrt{[d_{\mathrm{arc}}(\mathbf{q}_0,\, \mathbf{q}_{\mathrm{rot}_\Delta})]^2 + \|\, \mathbf{q}_{\mathrm{tra}_\Delta}\|^2}, \tag{3.35}$$

where $\mathbf{q}_0 = (1, 0, 0, 0)^\top$ denotes the zero rotation. Instead of d_{arc}, any of the quaternion metrics defined in Section 3.4.2 could be used and (3.35) would hold true defining a distance measure. We focus on distance (3.32) as it describes topologically correct quaternion divergence. The square root is introduced in the definition, as this allows for inserting d^2_{mag} into the kernel function without getting combined terms of rotational and translational parts.

Axis-angle Metric

We employ our findings in Section 3.4.2 and define the distance over rotations as the length of the geodesic between rotations, as the geodesic topologically correct describes the distance between rotations. This finding additionally is supportet by [86]. For rotation matrices and unit quaternions such arc distance functions are used by [87] and [46]. The arc distance for axis-angle can be derived from unit quaternions considering that the angle between unit quaternions is half the angle between rotations, because unit quaternions are a double coverage of the rotation group $SO(3)$. Hence, we obtain as distance function with the axis-angle representation exactly the double of the distance (3.32),

$$d_{\mathrm{geo}}(\theta_1 \mathbf{u}_1, \theta_2 \mathbf{u}_2) = 2\,\mathrm{acos}\left|\cos\frac{\theta_1}{2}\cos\frac{\theta_2}{2} + \sin\frac{\theta_1}{2}\,\sin\frac{\theta_2}{2}\,\mathbf{u}_1^\top \mathbf{u}_2\right|. \tag{3.36}$$

Proposition 3.4.2 *The distance measure* (3.36) *defines a metric.*

Proof 3.4.2 (Proposition 3.4.2) *The distance* (3.36) *is non-negative by definition of the inverse cosine. It is symmetric as the scalar product, the sum and the vector dot product are well-known to be symmetric. To proof the triangle inequality, we argue geometrically: The distance* (3.36) *is a valid parametrization of the geodesic length between rotations* $rot_1, rot_2 \in SO(3)$. *Hence, it is equivalent to show that*

$$d_{\mathrm{geo}}(rot_1, rot_2) + d_{\mathrm{geo}}(rot_2, rot_3) \geq d_{\mathrm{geo}}(rot_1, rot_3). \tag{3.37}$$

A proof for the geodesic length between rotations to define a metric is provided in [86]. ☐

Axis-angle and Translation Vector Distance Measure

As distance function on $SE(3)$ parametrized by axis-angle and translation vector we define the root over a sum of squares, inspired by the distance measure over dual quaternions introduced in (3.35). Additionally, we include in the distance function a convex combination of weights, $\sum_i \rho_i = 1$, where $\rho_i \geq 0$, to allow an application dependent scaling between rotation and translation, as we know from [87] that any distance metric in $SE(3)$ will ultimately depend on the choice of length scale. Hence, we define as distance function for rigid motions

$$d_{\text{wmag}}(\mathfrak{g}_1, \mathfrak{g}_2) = \sqrt{\rho_1 \left[d_{\text{geo}}(\theta_1 \mathbf{u}_1, \theta_2 \mathbf{u}_2)\right]^2 + \rho_2 \| \mathbf{v}_1 - \mathbf{v}_2 \|^2}. \tag{3.38}$$

This distance function (3.38) allows for incorporation of domain knowledge. To give an intuitive example, one can imagine the divergence in motion dynamics of a spinning top and a large truck. While the cost for a 2π rotation of a spinning top may be in the range of millimeters in translation, a truck may cover large distances easier than a full rotation. This dynamics difference should be reflected in the similarity measure between poses, as application specific weighting of rotation and translation increases the regression performance of the GP.

3.4.3 Mean and Kernel Functions

In the following we introduce a set of valid mean and kernel functions defined on the special Euclidean group $SE(3)$. For approximating dynamics of the form (2.4) we know from (3.1) that each mean and kernel function has the form

$$\begin{aligned} m &\colon SE(3) \to \mathbb{R} \\ k &\colon SE(3) \times SE(3) \to \mathbb{R}_0^+. \end{aligned} \tag{3.39}$$

For the mean function $m(\cdot)$ no further requirements need to be satisfied. In the special Euclidean group, the zero mean maps any input $\mathfrak{g} \in SE(3)$ to the zero element, $m(\mathfrak{g}) \equiv 0$. Kernel functions encode the correlation between the elements of the dynamics input domain $SE(3)$ and constitute the essential part of GP modeling. A kernel is required to be symmetric and positive semi-definite to define a valid kernel function. As in literature various strategies are available to construct more elaborate kernel functions by composition from elementary kernels, e.g. see [88], we focus on introducing one representative kernel per kernel class; specifically a stationary and a dot-product kernel.

Dot-product kernel

From [43] we know that the dot-product kernel defines a valid kernel function for elements $\mathbf{x}_i, \mathbf{x}_j \in \mathbb{R}^n, n \in \mathbb{N}, \forall i, j$. In our setting, we seek to approximate rigid-body motion dynamics, where the rigid motions $\mathfrak{g} \in SE(3)$. In Section 3.3 we have presented the dot products (3.21) and (3.29) for the parametrizations dual quaternions and axis-angle pseudo-vectors concatenated with Euclidean translation vectors, respectively. We have seen that in the respective spaces $\mathfrak{g} \equiv \mathbf{q}_{\text{re}} + \epsilon\, \mathbf{q}_{\text{du}} \in S_3/\{\pm 1\} \rtimes \mathbb{R}^3$ and $\mathfrak{g} \equiv (\theta \mathbf{u}, \mathbf{v}) \in \tilde{B}_\pi(0) \times \mathbb{R}^3$, the dot product describes the same operation as the vector space dot product in $\mathbb{R}^6$. Thus it follows,

$$k_{\text{dot}}(\mathfrak{g}_i, \mathfrak{g}_j) = \langle \mathfrak{g}_i, \mathfrak{g}_j \rangle + \sigma_n^2 \delta_{ij}, \tag{3.40}$$

with hyperparameter $\sigma_n^2 \geq 0$, defines a valid kernel function on the rigid motions $\forall i, j$.

Stationary kernel

One of the most important stationary kernels is the squared exponential kernel. In the following we show that the distance functions introduced in Section 3.4.2 indeed induce valid generalized squared exponential kernel functions.

To support an intuitive understanding, we explain the squared exponential kernel generalization (1) over pure rotations parametrized by unit quaternions S_3, (2) over full rigid motions parametrized by dual quaternions $\mathbb{H}_D$, and finally (3) over full rigid motions parametrized by axis-angle and translation vector.

Kernel over Unit Quaternions Over the metrics from Section 3.4.2, kernel functions on S_3 can be defined. We focus on the kernel function resulting from the arc length metric, as the underlying metric preserves the unit quaternion topology.

Theorem 3.4.1 *A valid kernel function* $k_{\text{se:arc}} : S_3 \times S_3 \to \mathbb{R}_0^+$ *over unit quaternions* $\mathbf{q},\ \mathbf{q}' \in S_3$*, is defined by*

$$k_{\text{se:arc}}(\mathbf{q},\,\mathbf{q}') = \sigma_f^2\ \exp\left(-\frac{[d_{\text{arc}}(\mathbf{q},\,\mathbf{q}')]^2}{2\lambda^2}\right) + \sigma_n^2 \delta_{ij}, \tag{3.41}$$

where d_{arc} *as in* (3.32) *and the hyperparameters* $\sigma_f, \lambda, \sigma_n > 0$.

To prove Theorem 3.4.1 we need to show that $k_{\text{se:arc}}(\mathbf{q},\,\mathbf{q}')$ is symmetric and positive definite $\forall\,\mathbf{q},\,\mathbf{q}' \in S_3$. Since proving positive definiteness is not easy, we provide an alternative solution. It is equivalent to show Lemma 3.4.1, following below, to prove Theorem 3.4.1. Therefore, we need to define the notion of conditional positive definiteness (cpd). Then, we prove Lemma 3.4.1 and thereafter, the Theorem 3.4.1.

Definition 3.4.1 (Zhang et al. [89]) *Let* $\mathcal{X}$ *be a nonempty set. A real-valued symmetric function* $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ *is called a conditionally positive definite (cpd) kernel function, if and only if the Gram matrix* $\mathbf{K} \in \mathbb{R}^{\nu\times\nu}$ *satisfies* $\mathbf{c}^\top\mathbf{K}\mathbf{c} \geq 0$ *for any vector* $\mathbf{c} \in \mathbb{R}^\nu$ *with* $\mathbf{c}^\top\mathbf{1} = 0$*, where* $\mathbf{1} = (1, \ldots, 1)^\top \in \mathbb{R}^\nu$.

Lemma 3.4.1 *The squared negative distance function*

$$k_{\text{snd}}(\mathbf{q},\,\mathbf{q}') := -([d_{\text{arc}}(\mathbf{q},\,\mathbf{q}')]^2) \tag{3.42}$$

defines a cpd kernel function.

Proof 3.4.3 (Lemma 3.4.1) *Let the functions* $g_1 : [-1,1] \to [-\frac{\pi}{2},\, \frac{\pi}{2}]$ *given by* $g_1(x) = \pi/2 - \arccos(x)$ *and* $g_2 : [-1,1] \to [\frac{3}{2}\pi,\, \frac{\pi}{2}]$ *given by* $g(x) = \pi/2 + \arccos(x)$*. Let us assume for now that the functions* g_1 *and* g_2 *are kernels. Given the finite product of kernels defines a kernel [66], then also the function* $g : [-1,1] \to [-\frac{3}{4}\pi^2,\, \frac{1}{2}\pi^2]$ *defined by* $g(x) = g_1(x)g_2(x)$ *is also a valid kernel. Hence, we can rewrite* (3.42) *to*

$$k_{\text{snd}}(\mathbf{q},\,\mathbf{q}') = -\frac{\pi^2}{4} + g(\min(\langle\,\mathbf{q}, \pm\,\mathbf{q}'\rangle)). \tag{3.43}$$

From [89] we know that a constant $c \in \mathbb{R}$ *is cpd, a valid kernel is also cpd and the finite sum of cpd kernels is cpd. This induces, that* (3.43) *is cpd. Thus, the proof is complete,* *if we can prove that the functions* g_1 *and* g_2 *are valid kernels. According to [89, Theorem 4] we know that the Taylor series expansion of a function defined on the unit sphere, such as* $\arcsin(x)$*, defines a valid covariance function, iff all coefficients are non-negative. Reformulating* $g_1(x) = \arcsin(x)$*, the function can be expressed as infinite polynomial with non-negative coefficients via Taylor expansion. Likewise, the function* g_2 *can be reformulated in terms of the arcsin function,* $g_2(x) = \pi + \arcsin(-x)$*. As the sum of kernels defines a kernel function according to [66], we obtain that* g_1 *and* g_2 *both define valid kernels.* ☐

Proof 3.4.4 (Theorem 3.4.1) *If a kernel* k *is cpd, it follows that* $\exp(ck)$ *is a valid kernel function for all constants* $c > 0$ *according to [72]. Further, we know from [66] that any finite constant* $\tilde{c} > 0$ *is a positive definite kernel function and the product of valid kernels defines a positive definite kernel function. Together, this proves that the function* $k_{\text{se:arc}}(\,\mathbf{q},\,\mathbf{q}')$ *defines a positive definite kernel function for all* $\sigma_f,\ \lambda, \sigma_n > 0$*.* ☐

By introducing the distance measures (3.30) resp. (3.33) into the formula for the squared exponential kernel (3.3), we obtain kernels $k_{\text{se:norm}}$ and $k_{\text{se:}TS_{\mathbf{q}}}$, respectively, in the Euclidean space. They, too, allow for a simple GP generalization to 3-DoF rotations.

Kernel over Dual Quaternions Analogously to the GP model generalization over pure rotations, a squared exponential kernel function for rigid motions in $SE(3)$ is obtained from the proposed dual quaternion distance measure.

Theorem 3.4.2 *A valid kernel function* $k_{\text{se:mag}} : (S_3 \times \mathbb{R}^3) \times (S_3 \times \mathbb{R}^3) \to \mathbb{R}_0^+$ *over dual quaternions* $\mathfrak{g},\ \mathfrak{g}' \in S_3 \times \mathbb{R}^3$*, is defined by*

$$k_{\text{se:mag}}(\,\mathfrak{g},\,\mathfrak{g}') = \sigma_f^2\ \exp\left(-\frac{[d_{\text{mag}}(\,\mathfrak{g},\,\mathfrak{g}')]^2}{2\lambda^2}\right) + \sigma_n^2\delta_{ij}, \tag{3.44}$$

where d_{mag} *as in* (3.35) *and the hyperparameters* $\sigma_f, \lambda, \sigma_n > 0$*.*

Proof 3.4.5 (Theorem 3.4.2) *We reformulate the function* (3.44) *using the property* $\exp(a+b) = \exp(a)\exp(b)$*, as for each factor separately, it is simple to show, that the factor defines a kernel function. Thus, we obtain*

$$k_{\text{se:mag}}(\,\mathfrak{g},\,\mathfrak{g}') = \sigma_f^2\ \underbrace{\exp\left(-\frac{[d_{\text{arc}}]^2}{2\lambda^2}\right)}_{\star}\underbrace{\exp\left(-\frac{\|\,\mathbf{q}_{\text{tra}}\|^2}{2\lambda^2}\right)}_{\diamond} + \sigma_n^2\delta_{ij}. \tag{3.45}$$

The factor $\star$ *defines a valid kernel function according to Theorem 3.4.1 and the factor* $\diamond$ *describes a valid kernel of standard form* (3.3)*. To finalize the proof, we apply the kernel construction rules via multiplication and addition. These rules state that any finite product, respectively sum, of kernels defines a kernel, see [66, Chap. 6.2].* ☐

Kernel over Axis-angle and Translation Vector

Theorem 3.4.3 *The function* $k_{\text{se:wmag}} : SE(3) \times SE(3) \to \mathbb{R}_0^+$*,*

$$k_{\text{se:wmag}}(\,\mathfrak{g}_i,\,\mathfrak{g}_j) := \sigma_f^2\ \exp\left(-\frac{[d_{\text{wmag}}(\,\mathfrak{g}_i,\,\mathfrak{g}_j)]^2}{2\lambda^2}\right) + \sigma_n^2\delta_{ij}, \tag{3.46}$$

where we consider a distance function d defined by (3.38) and positive hyperparameters $\mathbf{h} = (\lambda, \sigma_f, \sigma_n)$, namely length-scale $\lambda > 0$, signal variance $\sigma_f > 0$ and signal noise $\sigma_n > 0$, defines a valid kernel function.

Proof 3.4.6 *We know that the finite sum of kernels provides a kernel [66, Chap. 6.2] and it follows from definition that $\sigma_n^2\delta_{ij} \geq 0$ defines a valid kernel. Hence, we can focus on the first summand in the reminder of this proof. Introducing (3.38) in the noise-free squared exponential kernel, we can rewrite (3.46) neglecting the noise term $\sigma_n^2\delta_{ij}$ as*

$$\sigma_f^2 \underbrace{\exp\left(-\frac{\rho_1[d_{\text{geo}}(\theta_i\mathbf{u}_i, \theta_j\mathbf{u}_j)]^2}{2\lambda^2}\right)}_{\bigstar}\underbrace{\exp\left(-\frac{\rho_2\|\,\mathbf{v}_i - \mathbf{v}_j\|^2}{2\lambda^2}\right)}_{\clubsuit}. \tag{3.47}$$

A proof for the standard squared exponential kernel $\clubsuit$ to be valid is provided in [66, Chap. 6.2]. As any kernel scaled by a positive constant and any finite product of kernels yields each a new valid kernel [88], it suffices to show that also $\bigstar$ defines a valid kernel, to complete the present proof. To formally show that $\bigstar$ is a kernel, we utilize the cpd property as described in Definition 3.4.1. A kernel k being cpd induces that $\exp(ck)$ defines a valid kernel function $\forall c > 0$ [89]. Hence, it suffices to show that the squared negative distance function

$$k_{\text{2snd}}(\theta\,\mathbf{u}, \theta'\mathbf{u}') := -\left([d_{\text{geo}}(\theta\,\mathbf{u}, \theta'\mathbf{u}')]^2\right) \tag{3.48}$$

defines a cpd kernel. To derive that, we transform the orientation representation to unit quaternions, obtained from the axis-angle representation by $\mathbf{q} = \pm(\cos\frac{\theta}{2}, \sin\frac{\theta}{2}\,\mathbf{u})$. Then, introducing (3.32) into (3.48), we obtain

$$k_{\text{2snd}}(\,\mathbf{q},\,\mathbf{q}') = -4\left(\operatorname{acos}\left|\mathfrak{Re}(\,\mathbf{q}\overline{\mathbf{q}}')\right|\right)^2, \tag{3.49}$$

where $\mathfrak{Re}$ denotes the real part of a quaternion and $\overline{\mathbf{q}}$ the quaternion conjugate.

In the following we regenerate the proof of Lemma 3.4.1 for the slightly modified distance function $2\operatorname{acos}\left|\mathfrak{Re}(\,\mathbf{q}\overline{\mathbf{q}}')\right| = 2d_{\text{arc}}(\,\mathbf{q},\,\mathbf{q}')$. Let us substitute $\mathfrak{z} := \left|\mathfrak{Re}(\,\mathbf{q}\overline{\mathbf{q}}')\right|$. It is well-known, that $\operatorname{acos}(\,\mathfrak{z}) = \frac{\pi}{2} - \operatorname{asin}(\,\mathfrak{z})$. Hence, we can rewrite

$$k_{\text{2snd}}(\,\mathbf{q},\,\mathbf{q}') = -\pi^2 + 4\operatorname{asin}(\,\mathfrak{z})\,(\pi - \operatorname{asin}(\,\mathfrak{z}))\,. \tag{3.50}$$

From [89] we know that any constant $c \in \mathbb{R}$ is cpd, a valid kernel always is also cpd and the finite sum of cpd kernels is cpd. Thus, if we can prove that both auxiliary functions $g_1(\,\mathfrak{z}) = \pi - \operatorname{asin}(\,\mathfrak{z})$ and $g_2(\,\mathfrak{z}) = \operatorname{asin}(\,\mathfrak{z})$ for $\mathfrak{z} \in [-1, 1]$, define valid kernels, it follows that (3.50) is cpd, and therewith (3.49) and (3.48), which would conclude the proof. In the remaining part of the proof, we analyze the auxiliary functions $g_1(\,\mathfrak{z})$ and $g_2(\,\mathfrak{z})$. Our aim is to apply [89, Theorem 4] stating that a function $\check{f}\,(\langle\,\mathbf{q},\,\mathbf{q}'\rangle)$ defined on the unit sphere, $|\,\mathbf{q}| = |\,\mathbf{q}'| = 1$, in an infinite dimensional Hilbert space is a valid kernel if and only if its Taylor series expansion has only non-negative coefficients. It is well known, that the coefficients of asin, written as infinite series, are non-negative, and that $-\operatorname{asin}(\,\mathfrak{z}) = \operatorname{asin}(-\,\mathfrak{z})$ holds. To cope with the minus in function $g_1(\,\mathfrak{z})$, we re-substitute $-\,\mathfrak{z}$. The real part of a quaternion product is calculated

$$\mathfrak{Re}(\,\mathbf{q}\overline{\mathbf{q}}') = \langle\pm\,\mathbf{q}, \overline{\mathbf{q}}'\rangle. \tag{3.51}$$

Taking advantage of the fact that opposing unit quaternions $\pm\mathbf{q}$, *represent the same orientation,* $\pm\mathbf{q}$ *and* $\overline{\mathbf{q}}'$ *can be chosen to lie on the same hemisphere of* S_3, *such that we obtain a positive value for* $\langle \pm\mathbf{q}, \overline{\mathbf{q}}' \rangle$. *Hence, we are allowed to apply [89, Theorem 4] and the proof is complete.* □

3.5 Framework Completion through Integration from Velocity Space to Rigid Motions

In the proposed GP generalization, we seek to approximate a dynamics as given in (2.4), mapping from the rigid motions into the velocity space. In many control applications, however, it is of interest to generate a prediction of the motion including uncertainty certificates in $SE(3)$. This requires the integration of the velocity signal and the corresponding transformation of uncertainty, which is not trivial in general, as the space of rigid motions $SE(3)$ is non-Euclidean as soon as rotation is present. Consequently, the Gaussian describing the uncertainty in velocity space, has to be transformed to a curved space, where probability distributions inherently are not defined. This section discusses a completion of the GP framework for approximating rigid motion dynamics by presenting integration methods for both employed rigid motion representations.

3.5.1 Dual Quaternions

We present a projection from Euclidean space to the sphere S_3, which is used to define a probability distribution on the rigid motions and as integrator from velocity space $TSE(3)$ to $SE(3)$. As the representative $T_{\mathfrak{g}}SE(3)$ of the velocity space depends on the tangent point $\mathfrak{g}$ and this tangent point continuously changes with the motion, we additionally present how the representing tangent space $T_{\mathfrak{g}}SE(3)$ can be altered to a different representative $T_{\mathfrak{g}'}SE(3)$.

Projected Gaussian Distribution on $SE(3)$

Due to defining input/output spaces in form of dynamics (2.4), the generalized GP model provides the uncertainty prediction as a Gaussian distribution in velocity space $TSE(3)$. The dynamics formulation in discrete time, in contrast, maps from state space to state space,

$$\check{f}_{M_{\mathrm{d}}} : SE(3) \times T_{\mathfrak{g}}SE(3) \to SE(3). \tag{3.52}$$

As we have explained in Remark 2.1.3, it is not possible to model discrete-time dynamics on $SE(3)$ using the classical formulation of a mapping from $\mathfrak{g}_\kappa$ to $\mathfrak{g}_{\kappa+1}$ (because $SE(3)$ does not provide the required vector space structure). However, we provide here a method to transform the predictive Gaussian uncertainty estimate from the velocity space $TSE(3)$ into $SE(3)$.

We start with providing a probability distribution on pure rotation quaternions, visualized in Figure 3.2, by employing a projection from the tangent space onto the unit sphere S_3. Let $TS_{\mathbf{q}_{\mathrm{rot}}}$ be the tangent space of $\mathbf{q}_{\mathrm{rot}}$ to the unit hypersphere S_3. The central projection $\Pi_{\mathbf{q}_{\mathrm{rot}}}$ propagates a 3D Gaussian distribution $\mathcal{N}(\mu_{\mathrm{rot}}, \Sigma_{\mathrm{rot}})$ given in tangent space $TS_{\mathbf{q}_{\mathrm{rot}}}$ to the

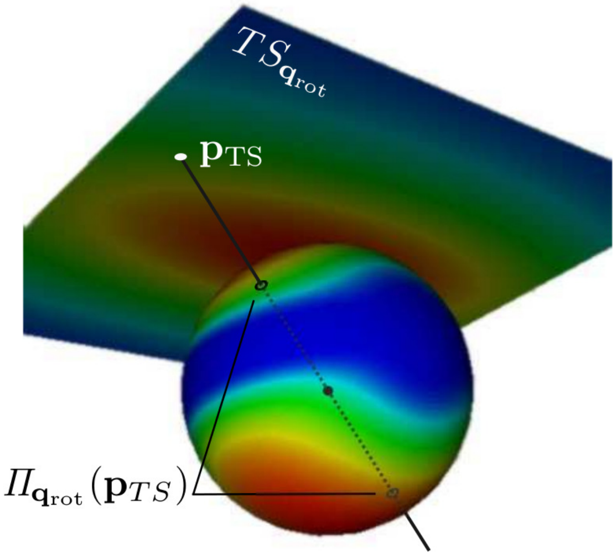


Figure 3.2: The Gaussian distribution in the tangent plane to the north pole is projected to the sphere S_2 via central projection. Hence, each point in the plane is projected to both opposite intersections of the sphere with a straight line through the center of the sphere and the point in the tangent plane.

sphere S_3 via

$$\begin{aligned}
&\Pi_{\mathbf{q}_{\mathrm{rot}}} : TS_{\mathbf{q}_{\mathrm{rot}}} \to S_3/\{\pm 1\} \\
&\Pi_{\mathbf{q}_{\mathrm{rot}}}(\mathbf{p}_{TS}) = \pm \frac{\mathbf{p}_{\mathbf{B}}}{\|\mathbf{p}_{\mathbf{B}}\|},
\end{aligned} \tag{3.53}$$

where $\mathbf{p}_{\mathbf{B}} = \mathbf{q}_{\mathrm{rot}} + \mathbf{B}\,\mathbf{p}_{TS} \in \mathbb{R}^4$ and $\mathbf{p}_{TS} \in \mathbb{R}^3$ any point in the tangent space $TS_{\mathbf{q}_{\mathrm{rot}}}$.

Lemma 3.5.1 *The projection's inverse* $\Pi^{-1}_{\mathbf{q}_{\mathrm{rot}}}(\mathbf{q}')$ *is calculated analytically using trigonometric calculus and the intercept theorem* $\forall\, \mathbf{q}' \in S_3,\ \mathbf{q}' \not\perp \mathbf{q}_{\mathrm{rot}}$.

The basis $\mathbf{B} = \{\boldsymbol{\beta}_1, \boldsymbol{\beta}_2, \boldsymbol{\beta}_3\} \in \mathbb{R}^{4\times 3}$ is the canonical representation of the 3D tangent space $TS_{\mathbf{q}_{\mathrm{rot}}}$ in the space $\mathbb{R}^4$.

Remark 3.5.1 *The stereographic projection is an alternative to the introduced central projection* (3.53)*, which uniquely maps all points on a unit sphere* $S_n, n \in \mathbb{N}+$ *to a certain tangent space. In contrast to projection* (3.53)*, where the center of projection is the sphere's center, the stereographic projection maps the points via a projection, which is centered on the opposite point of the tangent space. This means, if the tangent point is the north pole, the center of projection is the south pole.*

Remark 3.5.2 *The exponential function* $\exp(\cdot)$ *constitutes another alternative. In contrast to the central and stereographic projection, this mapping is length preserving, i.e. a distance in the velocity space equals the projected distance on the orientations. A disadvantage that correlates with preserving the distance, is the ambiguity of the projection. All points in the tangent space on a straight line through the tangent point, which have the distance* $2\pi k + d$ *to the tangent point, for a fixed* $d \in [0, 2\pi)$ *and an arbitrary* $k \in \mathbb{N}$*, are projected to the same point on the sphere.*

When projecting the Gaussian density φ_{TS} of the distribution $\mathcal{N}(\mu_{\mathrm{rot}}, \Sigma_{\mathrm{rot}})$ to the sphere $S_3/\{\pm 1\}$ using (3.53), its properties are inherited, as (3.53) is a bijective mapping (see Lemma 3.5.1). To obtain a probability density on the hypersphere, we normalize over the new domain $S_3/\{\pm 1\}$,

$$\varphi_{S_3} = \frac{1}{C}\varphi_{TS}, \text{ where } C = \int_{S_3/\{\pm 1\}} \varphi_{TS}\left(\Pi^{-1}_{\mathbf{q}_{\mathrm{rot}}}(\mathbf{q})\right) d\,\mathbf{q}, \tag{3.54}$$

following previous investigations in [45] and project the set closure $\{\infty\}$ of $TS_{\mathbf{q}}$ to the set of orthogonal vector-valued points $\mathbf{q}^{\perp}_{\mathrm{rot}}$ on the sphere,

$$\Pi_{\mathbf{q}_{\mathrm{rot}}}(\infty) = \mathbf{q}^{\perp}_{\mathrm{rot}} := \{\mathbf{q}' \in S_3 | \, \mathbf{q}' \perp \mathbf{q}_{\mathrm{rot}}\}. \tag{3.55}$$

As in the tangent space the set closure $\{\infty\}$ has probability 0, we obtain a continuous completion of the projected probability distribution on the sphere. The projected probability distribution is denoted by $\mathcal{N}(\mathbf{q}_{\mathrm{rot}}, \mu_{\mathrm{rot}}, \Sigma_{\mathrm{rot}})$.

Remark 3.5.3 *The central projection satisfies the desired property of mapping the Gaussian distribution on each hemisphere: opposing quaternions receive the same probability value. This fits exactly the topology of unit quaternions, as opposing unit quaternions represent the same orientation. Consequently, the positive probability on the sphere of distribution* $\mathcal{N}(\mathbf{q}_{rot}, \mu_{rot}, \Sigma_{rot})$ *does not reach a* π *rotation from the tangent point* $\mathbf{q}_{rot}$.

Remark 3.5.4 *In case a matrix* $\mathbf{R} \in SO(3)$ *is chosen as rotation representation, the stereographic projection from tangent space* $TS_{\mathbf{p_R}}$ *to the unit sphere* S_2 *constitutes the equivalent to the central projection [90]. The stereographic projection maps the Gaussian distribution uniquely onto the whole sphere* S_2 *except for the one point: the point* $-\mathbf{p_R}$ *opposing the tangent point* $\mathbf{p_R}$. *A continuous completion is in this case obtained by mapping the set closure* $\{\infty\}$ *with probability 0 to the point* $-\mathbf{p_R}$.

Next, we extend the projection (3.53) to full 6-DoF rigid motions. As Prop. 3.3.1 holds, defining a Gaussian counterpart on $S_3/\{\pm 1\} \rtimes \mathbb{R}^3$ is equivalent to defining one on $SE(3)$. Further, an isomorphism $T_{\mathfrak{g}}SE(3) \simeq TS_{\mathbf{q}_{\mathrm{rot}}} \times \mathbb{R}^3$ is induced by (3.17), for $\mathfrak{g}$ representing $(\mathbf{q}_{\mathrm{rot}}, \mathbf{q}_{\mathrm{tra}})$. Thus, the full projection is defined by

$$\begin{aligned} &\Pi_{\mathfrak{g}} : T_{\mathfrak{g}}SE(3) \to S_3 \times \mathbb{R}^3 \\ &\Pi_{\mathfrak{g}}\big((\mathbf{p}_{TS}, \mathbf{p}_{\mathbb{R}^3})\big) = \begin{cases} \Pi_{\mathbf{q}_{\mathrm{rot}}}(\mathbf{p}_{TS}), & \text{for } \Pi_{\mathfrak{g}}\big|_{TS_{\mathbf{q}_{\mathrm{rot}}}} \\ \mathbf{q}_{\mathrm{tra}} + \mathbf{p}_{\mathbb{R}^3}, & \text{for } \Pi_{\mathfrak{g}}\big|_{\mathbb{R}^3}. \end{cases} \end{aligned} \tag{3.56}$$

A distribution on the rigid motion space $S_3/\{\pm 1\} \rtimes \mathbb{R}^3$ is obtained by a regular concatenation of the projected Gaussian $\mathcal{N}(\mathbf{q}_{\mathrm{rot}}, \mu_{\mathrm{rot}}, \Sigma_{\mathrm{rot}})$ with a standard Gaussian $\mathcal{N}(\mu_{\mathrm{tra}}, \Sigma_{\mathrm{tra}})$ in the Euclidean space of translations $\mathbb{R}^3$,

$$\mu = \begin{pmatrix} \mu_{\mathrm{rot}} \\ \mu_{\mathrm{tra}} \end{pmatrix}, \; \Sigma = \begin{pmatrix} \Sigma_{\mathrm{rot}} & 0 \\ 0 & \Sigma_{\mathrm{tra}} \end{pmatrix}. \tag{3.57}$$

An analogous approach has been performed previously in [45].

Excursion: Alternative Distributions on the Sphere

Alternatively to the projected probability distribution $\mathcal{N}(\mathbf{q}_{\text{rot}}, \mu_{\text{rot}}, \Sigma_{\text{rot}})$ there are other probability distributions, which are defined on some unit sphere $S_n, n \in \mathbb{N}^+$. We briefly review the characteristics and limits of the most popular ones.

The most straight forward modification of the univariate normal distribution to fit the unit sphere S_1, is by wrapping the probability density around the sphere infinitely many times. The resulting probability distribution is called wrapped normal distribution and describes the circular univariate normal distribution. Its density is defined by

$$\varphi_{w_{S_1}}(\theta|\mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} \sum_{k=-\infty}^{\infty} \exp\left(\frac{-(\theta - \mu + 2\pi k)^2}{2\sigma^2}\right), \tag{3.58}$$

where μ and σ the mean and the standard deviation of the univariate normal distribution in flat space, respectively. The main shortcoming of this representation is the infinite sum of point wise probability densities, which cannot be calculated explicitly. Additionally, the generalization to wrapping a multivariate Gaussian distribution makes calculations complicated, since the wrapping has to be performed in all dimensions simultaneously.

On the unit sphere S_1 the von Mises distribution approximates the circular univariate normal distribution. Its density is defined by

$$\varphi_{S_1}(\theta|\mu, \kappa_\sigma) = \frac{\exp(\kappa_\sigma \cos(\theta - \mu))}{2\pi I_0(\kappa_\sigma)}, \tag{3.59}$$

where θ is a rotation angle, μ is the measure of location, κ_σ is the measure of concentration and $I_0(\cdot)$ is the modified Bessel function of order 0. The parameters μ and $1/\kappa_\sigma$ are equivalences of the mean μ and variance σ in the univariate normal distribution $\mathcal{N}(\mu, \sigma)$. In this distribution the infinite sum is replaced by the cosine. A generalization to higher dimensions is not straight forward.

The Fisher distribution [91] generalizes the von Mises distribution to the sphere S_2. It approximates a certain type of bivariate Gaussian distribution on S_2. However, it can only handle isotropic Gaussian distributions in the 2-dimensional plane. This means, the covariance matrix Σ has to have the form $\Sigma = \sigma\left(\begin{smallmatrix} 1 & 0 \\ 0 & 1 \end{smallmatrix}\right)$, and thus it describes a circle.

Through the von Mises-Fisher distribution, the isotropic Gaussian distributions with $n > 2$ dimension, was approximated on the sphere S_n. This distribution generalizes the von Mises distribution - analogously to the Fisher distribution - to higher dimensions $n \in \mathbb{N}^+$.

The restriction to isotropic Gaussians could not be overcome until 1982, when Kent introduced the Fisher-Bingham distribution [92], which later was named Kent distribution. For a vector $\mathbf{x} \in S_2$ the Kent density is defined by

$$\varphi_{S_2}(\mathbf{x}|\gamma_\mu, \kappa_\sigma, \beta_\sigma) = \frac{1}{C(\kappa_\sigma, \beta_\sigma)} \exp\left(\kappa_\sigma \gamma_{\mu_1} \mathbf{x} + \beta_\sigma \left((\gamma_{\mu_2} \mathbf{x})^2 - (\gamma_{\mu_3} \mathbf{x})^2\right)\right), \tag{3.60}$$

where $\gamma_\mu = (\gamma_{\mu_1}, \gamma_{\mu_2}, \gamma_{\mu_2})$ is a 3×3 orthogonal matrix, κ_σ is the measure of concentration, β_σ is the measure of ellipticity and $C(\kappa_\sigma, \beta_\sigma)$ is the re-normalization constant. The re-normalization is given by

$$C(\kappa_\sigma, \beta_\sigma) = 2\pi \sum_{k=0}^{\infty} \frac{\Gamma(k + 1/2)}{\Gamma(k + 1)} \beta_\sigma^{2k} \left(\kappa_\sigma/2\right)^{-2k-1/2} I_{2k+1/2} + (\kappa_\sigma), \tag{3.61}$$

where Γ is the gamma distribution and $I_{2k+1/2}(\cdot)$ is the modified Bessel function of order $2k + 1/2$. Since the re-normalization constant (3.61) involves an infinite sum, working with the Kent distribution becomes unfeasible. Additionally, the generalization to higher dimensions of this distribution is limited by the sophisticated re-normalization (3.61).

Integration from Tangent Bundle to Rigid Motions

In this section we discuss the use of the full projection (3.56) as integrator from the velocity space to the special Euclidean group $SE(3)$. We show that projection (3.56) composed with our dynamics formulation (2.4) provides the standard dynamics (3.52), $\check{f} = \Pi_{\mathfrak{g}} \circ \tilde{f}_{M_{\mathrm{d}}}$.

It is well-known that rotational velocities $\boldsymbol{\omega}$ are obtained from quaternions $\mathbf{q}_{\mathrm{rot}}$ by

$$\boldsymbol{\omega}(t) = 2 \frac{d\, \mathbf{q}_{\mathrm{rot}}(t)}{dt}\, \mathbf{q}_{\mathrm{rot}}(t). \tag{3.62}$$

However, in many practical applications the explicit form of $\mathbf{q}_{\mathrm{rot}}(t)$ is unknown. Thus, we propose to approximate the rotational velocity $\boldsymbol{\omega}$ by $\boldsymbol{\omega}_{\Delta} = \boldsymbol{\omega}\Delta t$, obtained by zero-order hold discretization with constant time intervals $\Delta t > 0$. For a time-discrete form of dynamics (2.4), the output at a certain time instance $t > 0$ becomes $(\,\mathfrak{g}(t),\, \mathbf{v}_{\mathfrak{g}}(t)) \in TSE(3)$. Employing isomorphism $T_{\mathfrak{g}}SE(3) \simeq TS_{\mathbf{q}_{\mathrm{rot}}} \times \mathbb{R}^3$, the tangent space velocity $\mathbf{v}_{\mathfrak{g}} = (\,\mathbf{v}_{TS},\, \mathbf{v}_{\mathbb{R}^3})$ is decomposed into a rotation and a translation part and the relation between rotational tangent space velocity and the discrete-time rotational velocity becomes apparent

$$\boldsymbol{\omega}_{\Delta} = 2\ \mathbf{v}_{TS}. \tag{3.63}$$

The transforming unit quaternion $\mathbf{q}_{\Delta}$, rotating from t-th to $(t + \Delta t)$-th orientation, can be calculated using the exponential function as integrator of the discrete time rotational velocity $\boldsymbol{\omega}_{\Delta}$,

$$\mathbf{q}_{\Delta} = \exp\left(\frac{\boldsymbol{\omega}_{\Delta}}{2}\right). \tag{3.64}$$

For integrating directly to $\mathfrak{g}(t+\Delta t)$ however, we propose to employ the full projection (3.56)

$$\Pi_{\mathfrak{g}}\Big((\,\mathbf{v}_{TS},\, \mathbf{v}_{\mathbb{R}^3})\Big) = (\,\mathbf{q}_{\mathrm{rot}}(t + \Delta t),\, \mathbf{q}_{\mathrm{tra}}(t + \Delta t))\,, \tag{3.65}$$

where $\mathbf{q}_{\mathrm{rot}}(t + \Delta t) = \mathbf{q}_{\Delta}\, \mathbf{q}_{\mathrm{rot}}$ and $\mathbf{q}_{\mathrm{tra}}(t + \Delta t) = \mathbf{q}_{\mathrm{tra}} + \mathbf{v}_{\mathbb{R}^3}$. Thus, we spare the extra quaternion multiplication and after dual quaternion composition (3.19) we end up with the output of the standard dynamics (3.52).

Tangent Bundle Representative

The central projection introduced in (3.56) is a bijective mapping from any tangent space $T_{\mathfrak{g}}SE(3)$ to $SE(3)$. Its inverse $\Pi_{\mathfrak{g}}^{-1}$ is structurally equal to the back projection to any other tangent space $\Pi_{\mathfrak{g}'}^{-1}$. Therefore, we introduce in the following a machinery that provides the corresponding Gaussian in an arbitrary tangent space $T_{\mathfrak{g}'}SE(3)$.

The difficulty that arises is that on projecting each single value from the unit hypersphere to the new tangent space, no Gaussian is obtained any more, as can be seen in Figure 3.3. The original Gaussian (blue dashed) is defined in the tangent space $TS_{\mathbf{q}_{\mathrm{rot}}}$, the north pole. In the tangent space $TS_{\mathbf{q}'_{\mathrm{rot}}}$ the pointwise projected curve is visualized by the red solid line.

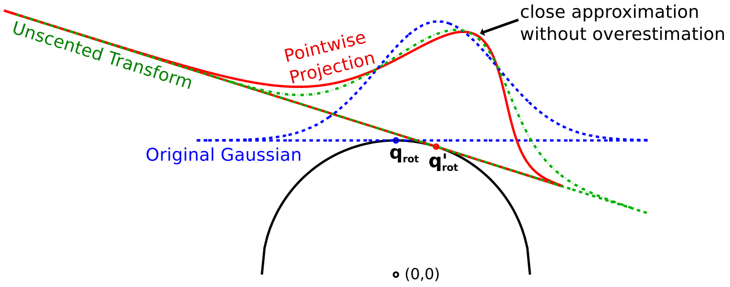


Figure 3.3: The Gaussian distribution depicted by the dashed blue line is projected via the sphere from the tangent space at $\mathbf{q}_{\mathrm{rot}} = (0, 1)$ into the tangent space at $\mathbf{q}'_{\mathrm{rot}} = (0.3, 0.95)$. After projection, the red curve is approximated using the unscented transform by a Gaussian, visualized by the green dashed-dotted line.

The skewness of the pointwise projection increases with the angle size between the tangent points $\mathbf{q}_{\mathrm{rot}}$ and $\mathbf{q}'_{\mathrm{rot}}$. For a Gaussian approximation of the pointwise projection, we propose to use the unscented transform [93]. Even though the unscented transform is well known from the unscented Kalman filter, it is a novel approach to apply it to GP modeling. For the unscented transform, sigma points ζ_l are calculated by

$$\zeta_{l,2n+l} = \mu_{\mathrm{rot}} \pm \left(\sqrt{(n+\varkappa_0)\Sigma_{\mathrm{rot}}}\right)_l \tag{3.66}$$

for $l = 1, \ldots, n$, where n is the dimension of the Gaussian, $\varkappa_0 \geq 0$ a scaling parameter (see details below) and $\zeta_0 = \mu_{\mathrm{rot}}$. They are projected to the tangent space $TS_{\mathbf{q}'_{\mathrm{rot}}}$ via

$$\Pi^{-1}_{\mathbf{q}'_{\mathrm{rot}}}(\Pi_{\mathbf{q}_{\mathrm{rot}}}(\zeta_l)). \tag{3.67}$$

In this new tangent space, the mean $\mu'_{\mathrm{rot}}(\varkappa_0)$ is given by the weighted average and the covariance $\Sigma'_{\mathrm{rot}}(\varkappa_0)$ by the weighted outer product of the transformed points [93]. The scaling parameter $\varkappa_0$ in the sigma points ζ_l is chosen such that the Kullback-Leibler divergence $D_{\mathsf{KL}}(P||Q)$ [94] between the approximated Gaussian $Q = \mathcal{N}(\mu'_{\mathrm{rot}}, \Sigma'_{\mathrm{rot}})$ and the pointwise projected distribution $P = \Pi^{-1}_{\mathbf{q}'_{\mathrm{rot}}}(\Pi_{\mathbf{q}_{\mathrm{rot}}}(\mathcal{N}(\mu_{\mathrm{rot}}, \Sigma_{\mathrm{rot}})))$ is minimized. For clarity in the presentation we substitute the expression

$$\mathfrak{z} := \Pi^{-1}_{\mathbf{q}_{\mathrm{rot}}}\left(\Pi_{\mathbf{q}'_{\mathrm{rot}}}(\mathbf{p}')\right) \tag{3.68}$$

and introduce $\mathbf{p}'$ as short notation for $\mathbf{p}_{TS_{\mathbf{q}'_{\mathrm{rot}}}}$. Then, $\varkappa_0 = \underset{\varkappa \in \mathbb{R}^+}{\arg\min}\, D_{\mathsf{KL}}(P||Q)$ is determined using

$$D_{\mathsf{KL}}(P||Q) = \int_{\mathbb{R}^n} (2\pi)^{-\frac{n}{2}} \left|\Sigma_{\mathrm{rot}}\right|^{-\frac{1}{2}} \exp(-g_1(\mathfrak{z})) \left(\ln \sqrt{\left|\frac{\Sigma_{\mathrm{rot}}}{\Sigma'_{\mathrm{rot}}}\right|} + \frac{1}{2}\left(g_1(\mathfrak{z}) - g_2(\mathbf{p}', \varkappa)\right)\right) d\,\mathbf{p}', \tag{3.69}$$

where the auxiliary functions g_1, g_2 are given by

$$\begin{aligned} g_1(\mathfrak{z}) &= \frac{1}{2}(\mathfrak{z} - \mu_{\mathrm{rot}})^\top \Sigma_{\mathrm{rot}}^{-1}(\mathfrak{z} - \mu_{\mathrm{rot}}) \\ g_2(\mathbf{p}', \varkappa) &= \frac{1}{2}(\mathbf{p}' - \mu'_{\mathrm{rot}}(\varkappa))^\top \Sigma'_{\mathrm{rot}}(\varkappa)^{-1}(\mathbf{p}' - \mu'_{\mathrm{rot}}(\varkappa)). \end{aligned} \tag{3.70}$$

The Gaussian Q, illustrated by the green dashed-dotted line in Figure 3.3, approximating the true underlying pointwise projected distribution P, visualized by the red solid line, meets the desires trade-off between over and under estimating the uncertainty in the probability distribution.

3.5.2 Axis-angle and Translation Vectors

In the following, we discuss how the exponential function is employed to integrate predictive angular velocities to the system states. $SO(3)$ is a compact Lie group with corresponding Lie algebra $\mathfrak{so}(3)$, the set of skew-symmetric matrices

$$\mathfrak{so}(3) = \left\{ \Omega \in \mathbb{R}^{3\times3} \middle| \Omega^\top = -\Omega \right\}. \tag{3.71}$$

It is well-known, see [95, Chap. 3.7], that for Lie groups the exponential map

$$\exp : \mathfrak{so}(3) \to SO(3) \tag{3.72}$$

pulls back elements to the special Orthogonal group. Given any angular velocity vector $\boldsymbol{\omega} = (\omega_\phi, \omega_\psi, \omega_\vartheta)^\top \in \mathbb{R}^3$, where $\theta_\omega = \|\boldsymbol{\omega}\|$ and $\mathbf{u}_\omega = \frac{\boldsymbol{\omega}}{\|\boldsymbol{\omega}\|}$, the corresponding skew-symmetric matrix

$$[\boldsymbol{\omega}]_\times = \begin{pmatrix} 0 & \omega_\phi & \omega_\psi \\ -\omega_\phi & 0 & \omega_\vartheta \\ -\omega_\psi & -\omega_\vartheta & 0 \end{pmatrix} \tag{3.73}$$

is pulled back to an element $\boldsymbol{R}$ in the special Orthogonal group $SO(3)$ using

$$\boldsymbol{R} = \exp([\theta_\omega \mathbf{u}_\omega]_\times). \tag{3.74}$$

From [86] we know that (3.74) can be calculated using Rodrigues' rotation formula

$$\boldsymbol{R} = \cos\theta_\omega \boldsymbol{I} + \sin\theta_\omega [\,\mathbf{u}_\omega]_\times + (1 - \cos\theta_\omega)\,\mathbf{u}_\omega\,\mathbf{u}_\omega^\top, \tag{3.75}$$

where $\boldsymbol{I}$ denotes the identity matrix. This induces that in axis-angle space the exponential function acts locally as identity on the angular velocities $\boldsymbol{\omega} \in \mathfrak{so}(3)$, what formally means $\exp(\boldsymbol{\omega}) = \exp(\theta_\omega \mathbf{u}_\omega) = \theta_\omega \mathbf{u}_\omega \in SO(3)$. Consequently, a Gaussian, representing the velocity prediction uncertainty, which is defined in the tangent space, describes equally the uncertainty over rotations in $SO(3)$ for a small neighborhood U around $\theta_\omega \mathbf{u}_\omega$. Outside the neighborhood the variance is still approximated by the Gaussian, but with increasing Euclidean distance, the approximation quality decreases rapidly. Hence, we can use the exponential function to integrate in discrete-time from the tangent bundle $TSE(3)$ to the special Euclidean group $SE(3)$. Thus, we obtain discrete-time dynamics by composing dynamics (2.5) with the integration, $\exp \circ f$.

3.6 Comparison of Gaussian Process Variants

In Section 3.2.3 we introduced the classic 6-DoF GP for rigid motion approximation. However, an application of $\mathrm{GP}_{\mathbb{R}^6}$ is reasonable solely if it can be guaranteed that all occurring angles are in close neighborhood to zero, where Euler angles are approximately Euclidean. Elsewhere, a generalized GP should be preferred to assure mathematical soundness of the model on $SE(3)$.

Both presented GP generalizations have good rights to exist in parallel. Their properties vary depending on the underlying rigid motion representation, as the topological structure of the rotation representation using unit quaternions on S_3 is different to the axis-angle pseudovector: In contrast to the rotation representation on a solid ball $\tilde{B}_\pi(0) \subset \mathbb{R}^3$, defined in (3.23), the unit quaternions lie on the unit hypersphere $S_3 \subset \mathbb{R}^4$. While the combination of a unit quaternion with a translation vector results in a dual quaternion, which represents the whole rigid motion in a unified manner, the axis-angle representation combined with translation vector is denoted by a pair. This separation can result in errors such as false application order of rotation and translation, as those do not commute. An advantage, however, lies in the direct access to the weighting factors ρ_1, ρ_2 of the distance function (3.38). Using the representation via dual quaternions, a weighting between rotation and translation is not directly feasible.

Further, the use of quaternions harbors the disadvantage of a higher probability of training failures due to the topology of quaternions that induces sensitive reaction to the hyperparameter choice. Even though both GP variants use geodesic distances, differences in the definition of the metrics result in different GP training behavior, as the distance function affects the correlation between rigid motions. In the experimental evaluations we demonstrate that the GP over axis-angle and translation vector exhibits in the training phase a more robust behavior towards the initial choice of hyperparameters.

The model training complexity of both generalized GP variants is equal to the standard GP complexity of $\mathcal{O}(n^3)$, where n is the number of training samples, which is due to inversion of an $n \times n$ covariance matrix. With a fixed training data size, the GP prediction complexity for all 3 variants is $\mathcal{O}(n)$, where n is the number of predicted points. In addition, we analyze the computational demand by counting floating-point operations (flops). Given a training set of n samples, the required number of flops for calculating the covariance matrix using the squared exponential kernel function is $18n^2+36n+16$ for the standard GP, $24n^2+85n+18$ for the generalized GP over dual quaternions and $22n^2+26n+15$ for the generalized GP over axis-angle and translation vector, respectively. This results in significant runtime improvement, in particular as the number of training samples increases, which is demonstrated in the next section.

Another difference is that the topological structure of quaternions requires a more complex integration than the computationally more efficient GP generalization: In the GP over dual quaternions, the integration is handled by an operation that projects values from the velocity space to the manifold of rigid motions $SE(3)$. In addition, a computationally demanding tangent space change operation is necessary, to disambiguate the tangent bundle representative, which becomes redundant in the generalized GP over axis-angle and Euclidean vector, as the integration via exponential function locally acts as identity, see (3.74) and below. This approach, however, represents uncertainty on a locally flat neighborhood around the estimation only, but does not allow for representing uncertainty on $SE(3)$ globally. Hence, the robust and efficient GP generalization is not suitable for representing wide spread uncertainty on the manifold $SE(3)$.

The choice of the generalized GP model depends on the aspired application properties and cannot be answered in general.

3.7 Experimental Evaluation

We start with a numerical evaluation of the benefits of the GP generalization in contrast to the standard 6-DoF GP. We exemplary employ the rigid motion representation dual quaternions in this demonstration. Then, we provide a real data experiment evaluating the influence of the underlying distance function between unit quaternions on the GP prediction, before we present a real data experiment employing the axis-angle and translation vector representation in the GP generalization. Finally, we compare the runtime and the rotation prediction accuracy of both introduced GP generalizations with the classic 6-DoF GP. For easier differentiation we denote in the following the classic 6-DoF GP by $\mathrm{GP}_{\mathbb{R}^6}$, the generalized GP over unit and dual quaternions by $\mathrm{GP}_{\mathbb{H}_D}$ and the generalized GP oder axis-angle and translation vector by $\mathrm{GP}_{\tilde{B}}$.

All experiments were implemented in MATLAB using the GPML toolbox, available on `gaussianprocess.org` [96], on a commercially available PC with Intel core i5-6360U processor and 8 GB RAM. The GPML toolbox was extended to additionally comprise the generalized GP over axis-angle and translation vector and the GP over dual quaternions.

3.7.1 Numerical Evaluation of GP Generalization Benefits

In this section we validate the introduced framework for generalized GP modeling over rigid motions ($\mathrm{GP}_{\mathbb{H}_D}$) in comparison to the state of the art $\mathrm{GP}_{\mathbb{R}^6}$. Our evaluations confirm that our approach significantly outperforms the traditional way of including orientation into the GP via Euler angles. We carry out a numerical performance analysis on simulated data, in which we calculate the prediction error as a function of the rotation magnitude and of the number of training samples.

More precisely, we evaluate the prediction accuracy of $\mathrm{GP}_{\mathbb{H}_D}$ and $\mathrm{GP}_{\mathbb{R}^6}$ as a function of the sample size of the training data and of the rotation range in the dynamics drift term of ten synthetically generated ground truth dynamics of form (2.5). The evaluation shows the necessity to introduce the $\mathrm{GP}_{\mathbb{H}_D}$ over dual quaternions for approximating nonlinear dynamics of rigid motions, as it clearly outperforms the classic $\mathrm{GP}_{\mathbb{R}^6}$.

Experimental Conditions

We randomly generate ten dynamics $f_{M_\mathfrak{d}}$ of form (2.5) based on the trigonometric functions $\{\sin, \cos, \tan\}$, with constants in the range $[\frac{1}{8}, 5]$ and basal operations $\{\pm, \times, \div, \circ\}$. Each 6-DoF dynamics component comprises three to seven operations. As an example, one of the dynamics is given by

$$f_M(\mathfrak{g}) = \begin{pmatrix} c\left(\frac{\pi}{3} - \cos(\mathbf{q}_{\mathrm{re}_x}) + \sin(\mathbf{q}_{\mathrm{re}_y} + \mathbf{q}_{\mathrm{du}_x})\right) \\ c\,(1 - \mathbf{q}_{\mathrm{re}_x} \sin(\mathbf{q}_{\mathrm{du}_z} + \mathbf{q}_{\mathrm{du}_x} + \mathbf{q}_{\mathrm{re}_z})) \\ c\left(\frac{\pi}{8} + \sin(\mathbf{q}_{\mathrm{du}_y} \mathbf{q}_{\mathrm{re}_z} - 1)\right) \\ 2\cos(\mathbf{q}_{\mathrm{re}_x} \mathbf{q}_{\mathrm{re}_y}) \sin(\mathbf{q}_{\mathrm{du}_x} + \mathbf{q}_{\mathrm{re}_x}) + \mathbf{q}_{\mathrm{du}_x} \\ \sin(\tan(\frac{\mathbf{q}_{\mathrm{re}_z}}{3} + \frac{1}{2}) - 2\,\mathbf{q}_{\mathrm{re}_y}) \\ 3\cos(\mathbf{q}_{\mathrm{du}_y} \mathbf{q}_{\mathrm{du}_z}) - \frac{\mathbf{q}_{\mathrm{re}_y}}{4} \end{pmatrix}, \tag{3.76}$$

where $\mathfrak{g} = (\mathbf{q}_{\mathrm{re}}, \mathbf{q}_{\mathrm{du}})$ and $c > 0$ a constant. For each of the dynamics, we generate randomly sampled training data sets and a ground truth data set for evaluating the estimation accuracy. For reference, each data set is converted to the traditional representation $\mathfrak{g} = (\phi, \psi, \vartheta, v_x, v_y, v_z)^\top$ invoking (3.17). To assure numerical stability we add an artificial white Gaussian noise with arbitrarily selected standard deviation $\sigma = 3 \cdot 10^{-3}$ to the training data sets. We use the squared exponential kernels (3.3) respectively (3.44) in the GP models.

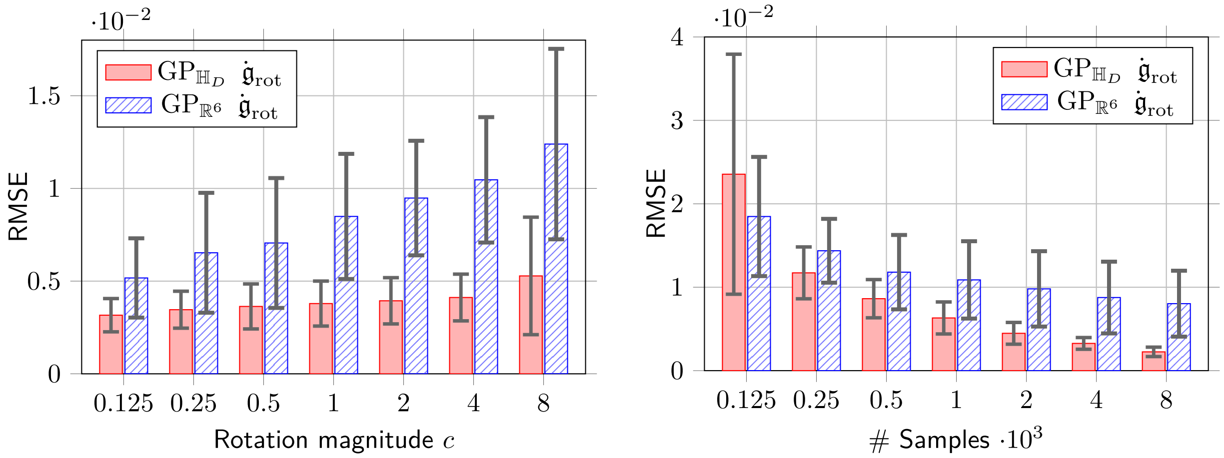
In order to analyse the prediction quality of the $\mathrm{GP}_{\mathbb{H}_D}$ and the $\mathrm{GP}_{\mathbb{R}^6}$, we employ the root-mean-square error (RMSE) between the predictive mean $\hat{\mathfrak{g}}$ and the true function value $f_{M_\mathrm{d}}(\mathfrak{g})$. For the translational output $f_{M_\mathrm{d}}(\mathfrak{g})\Big|_{\mathrm{tra}}$ we find no significant difference in the prediction quality of $\mathrm{GP}_{\mathbb{H}_D}$ and the classic GP in our experiments. For growing training data sets, the average RMSE in translational GP prediction converges around 10^{-3} for both models, the size of which corresponds to the magnitude of the noise with the difference in the RMSEs ranging in the magnitude of 10^{-4}. Therefore, we concentrate in the reminder of this evaluation on the quality of the rotational part of the predictive mean $\hat{\mathfrak{g}}_{\mathrm{rot}}$. In order to obtain reliable evaluation results, the RMSE is calculated for a test set $\{(\mathfrak{g}_i, f_M(\mathfrak{g}_i))\}_{i=1}^{10000}$ of iid pairs. The experiment is implemented in Matlab using the GPML toolbox [96]. For the hyperparameter optimization, we use the algorithm provided in the toolbox with a Gaussian likelihood. We initialize the training phase each time with the same set of default hyperparameters $H_0 = (\lambda_0, \sigma_{f0}, \sigma_{n0})$ for $\mathrm{GP}_{\mathbb{H}_D}$. As the signal range in $\mathrm{GP}_{\mathbb{R}^6}$ is π times the signal range in $\mathrm{GP}_{\mathbb{H}_D}$, we set the default starting hyperparameters in $\mathrm{GP}_{\mathbb{R}^6}$ to $(\lambda_0, \pi\sigma_{f0}, \pi\sigma_{n0})$.

RMSE Depending on Rotation Magnitude

In the first evaluation, we compare the prediction accuracy of the GP models, depending on the rotation magnitude in (3.76). We double the constant c in the dynamics f_{M_d} from 0.125 to 8 in a stepwise manner. The number of training samples is fixed to 2000 for all 70 (seven steps times ten dynamics) repetitions, and each time, the learning is performed per dynamics with the same set of uniformly drawn random samples. The average prediction RMSEs with standard deviation of both GP models are contrasted. In all tested conditions the $\mathrm{GP}_{\mathbb{H}_D}$ clearly outperforms the $\mathrm{GP}_{\mathbb{R}^6}$ prediction accuracy as visualized in Figure 3.4a. For small rotational velocities, the error difference between the models is less significant and the superiority of $\mathrm{GP}_{\mathbb{H}_D}$ may not counterbalance the additional computational complexity. For increased rotation magnitude the performance of the $\mathrm{GP}_{\mathbb{R}^6}$ is drastically decreased while the accuracy of the $\mathrm{GP}_{\mathbb{H}_D}$ is significantly less affected. The largest condition tested, $c = 8$, incorporated up to three full turns in the randomly generated dynamics. The corresponding rightmost result in Figure 3.4a indicates that the approximation abilities of $\mathrm{GP}_{\mathbb{H}_D}$ can be exhausted for even larger rotation magnitudes. At a certain magnitude the dynamics gets too edged for appropriate application of the squared exponential kernel.

RMSE Depending on Number of Samples

In the second numerical evaluation, we illustrate the GP prediction quality for $\mathrm{GP}_{\mathbb{H}_D}$ and the classic $\mathrm{GP}_{\mathbb{R}^6}$ for different numbers of training samples. The sample set size is relevant for the ability of a GP to capture the underlying function, as the stepwise orientation change increases for smaller sample sets. In this setup the rotation magnitude is fixed to $c = 1$. The model learning is performed on increasing sizes of training data sets, doubled in each step from 125 to 8000 samples. Figure 3.4b shows that the $\mathrm{GP}_{\mathbb{H}_D}$ approximates the



(a) Prediction quality for a fixed sample size of the training data and increasing rotation magnitude

(b) Prediction quality for increasing numbers of samples used in GP training and fixed rotation magnitude

Figure 3.4: Numerical performance analysis in predicting dynamics f_M. The bars depict the average RMSEs in the rotational velocity prediction, where solid red colors show the $\mathrm{GP}_{\mathbb{H}_D}$ and striped blue the classic GP. The standard deviation among the ten different dynamics is shown by the error bars.

underlying ground truth dynamics significantly better than the $\mathrm{GP}_{\mathbb{R}^6}$ except for the condition with 125 training samples. In the first tested condition, no significant difference could be determined by an independent-sampled t-test due to the large variance in the RMSE, $t(9) = 0.74, p = 0.47$. On the one hand, the results indicate that the $\mathrm{GP}_{\mathbb{H}_D}$ has difficulties finding optimal hyperparameters for very sparse sample sets. On the other hand, this evaluation demonstrates the superiority of $\mathrm{GP}_{\mathbb{H}_D}$ in approximating rigid motion dynamics involving significant rotation over $\mathrm{GP}_{\mathbb{R}^6}$ for sufficiently large sample sets.

3.7.2 Real World Experiment: GP Behavior depending on Unit Quaternion Distance

In this section a real data experiment is conducted to compare the estimation behavior of the generalized GP depending on the underlying distance metric over unit quaternions in the squared exponential kernel. The results are compared to the classic GP model incorporating Euler angles as rotation representation and the standard Euclidean distance.

Experimental Setup

A human participant draws repetitively 40 large “Z” letters on a flip chart. Even though not instructed to do so, the human naturally follows the pen with the eye. Using a combined system of motion capturing by Qualisys motion tracker (Sweden) and Dikablis eye tracking (Germany), the human gaze motion was captured at 100 Hz in the workspace coordinates. Figure 3.5 illustrates one experiment trial. From the measurements, we obtain approximate derivatives of the gaze motion using zero-order hold. We consider solely the head and gaze rotation, as we aim to compare *pure rotation approximation behavior* of the generalized GP

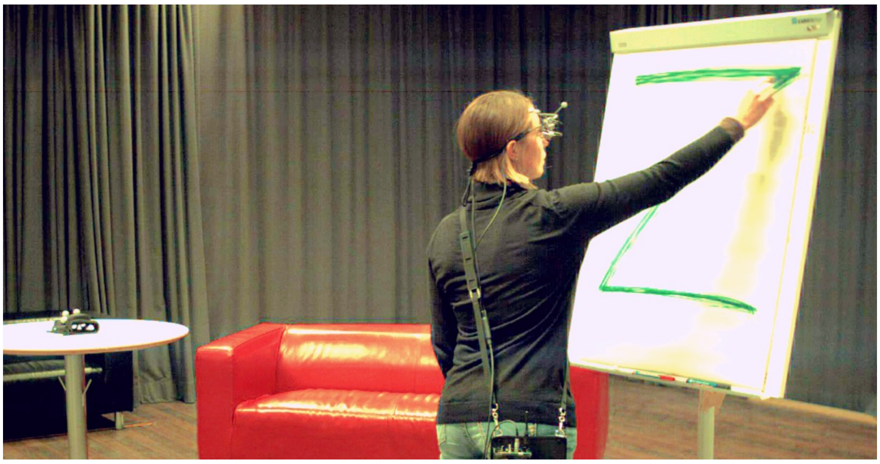


Figure 3.5: A human participant repetitively draws large “Z” letters on a flip chart, while naturally tracking the pen with the eye. The human gaze motion is captured in workspace coordinates using a combination of eye tracking and marker based head motion tracking.

describing human gaze dynamics of form (2.4). The captured data includes rotation up to 1.4 rad per time step, as gaze jumps in saccades, and involves a high noise level as the eye tracking system in our setup is nonlinearly delayed against the motion capture system. We intentionally select a challenging experimental setting to exhaust the GP approximation abilities as only borderline test cases bring up the behavioral differences. We train the GP models on the first 34 out of the 40 trials and test its performance on the remaining six repetitions. As in the numerical evaluation, we fix the initial hyperparameter values $\mathbf{h} = (\lambda_0, \sigma_{f0}, \sigma_{n0})$ for all GPs over unit quaternions and scale signal variance and signal noise about π for the reference GP over Euler angles.

Visualization Interpretation

Figure 3.6 illustrates the prediction behavior of the four compared GP models, namely Arc, Norm, $TS_{\mathbf{q}}$ and (ϕ, ψ, ϑ). The GP models we compare are the following: in first row a generalized GP using the arc metric (3.32) is shown, in second row the underlying metric is the quaternion norm (3.30), in third row the generalized GP is based on the tangent space distance (3.33). Fourth row illustrates the prediction behavior of the standard GP model over Euler angles using the Euclidean distance in the kernel function (3.3) for reference. More precisely, the first three rows visualize each the rotational velocity dimensions, $\{\dot{\mathfrak{g}}_{TS_x}, \dot{\mathfrak{g}}_{TS_y}, \dot{\mathfrak{g}}_{TS_z}\}$ of the tangent space to the sphere S_3 for the conditions Arc, Norm and $TS_{\mathbf{q}}$, while the fourth row visualizes the synchronized rotational velocities $\{\dot{\phi}, \dot{\psi}, \dot{\vartheta}\}$ for the Euler angles (ϕ, ψ, ϑ). For better visibility of the differences in the estimation we do not show all test trials of the “Z” repetitions in every column, but zoom in to the trials 1 and 2 in the first column, 3 and 4 in the second column and 5 and 6 in the third column. We visualize

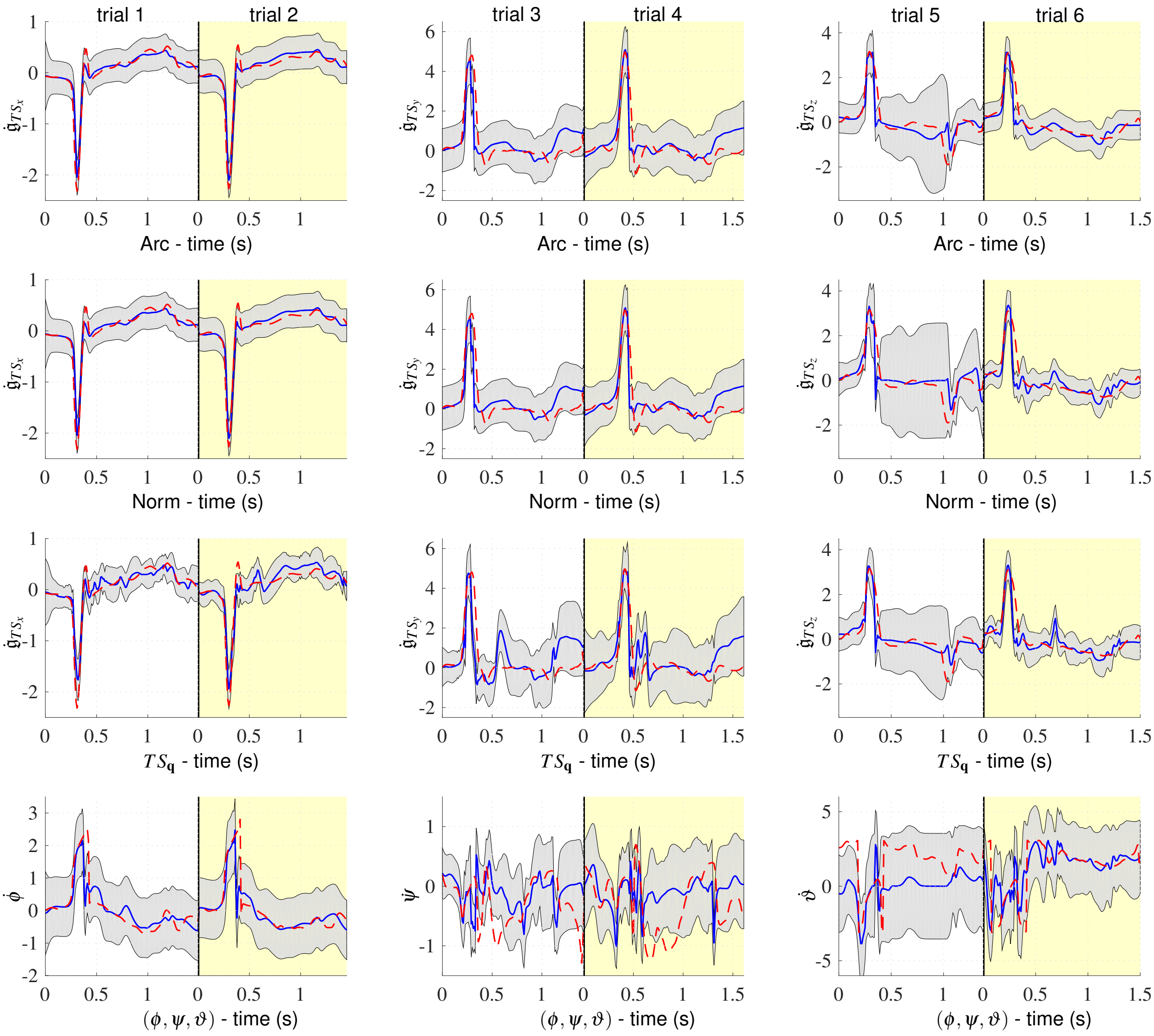


Figure 3.6: The prediction accuracy of the human gaze motion is illustrated. Blue solid lines show the mean prediction, red dashed lines the ground truth and gray shaded areas the prediction uncertainty. The columns correspond to the tangent bundle respectively rotational velocity dimensions, the rows to the different conditions. All columns are synchronized over time. In row one the GP model is based on the arc distance, in row two on the quaternion norm, in row three on the distance projected to the tangent space and in row four on the Euclidean distance in combination with the Euler angle representation. Column wise a second trajectory is highlighted by yellow background color.

the motion velocities over time in seconds and reset the runtime to zero at the beginning of every trial. Every second trial is contrasted per column by yellow background and the vertical black line in every plot indicates the beginning of that next trial. Per dimension and test condition (Arc, Norm, $TS_{\mathbf{q}}$ and (ϕ, ψ, ϑ)) we depict by a red dashed line the ground truth velocities, by a blue solid line the mean prediction and by a gray shaded area the 2σ-confidence region. Overall, the GP based on the arc metric (row one) outperforms the others, which can be seen most clearly in the third component of the velocity predictions. The squared exponential kernel over the quaternion norm overestimates the correlation of orientations slightly $\forall\, \mathbf{q},\ \mathbf{q}' \in S_3$. The kernel used in third row holds $k_{\text{se}:TS_{\mathbf{q}}}(\mathbf{q},\, \mathbf{q}') < k_{\text{se:arc}}(\mathbf{q},\, \mathbf{q}')$ $\forall\, \mathbf{q} \neq \mathbf{q}'$ in contrast. Therefore, it shows a much faster decrease in correlation for growing dissimilarity of the quaternions. In our illustrative example it forces the optimization to significantly smaller length scale values than in the conditions Arc and Norm. This behavior can be valuable in applications where highly precise orientation is essential. The generalized GP models have in common that all of them significantly outperform the classic GP model (condition (ϕ, ψ, ϑ)) which has problems capturing the large rotational velocities.

3.7.3 Real World Experiment: Generalized GP using Axis-angle and Translation Vector

In this experiment we evaluate the $\mathrm{GP}_{\tilde{B}}$ over axis-angle and translation vector in a real world application. Our goal is to learn human behavior from demonstration. Applications for reproduction and prediction of human-like movements are widespread in robotics, e.g. in programming by demonstration and assistive robotics. Consider a human-robot interaction scenario where a robot should help putting on a shoe to an elderly. Therefore, we require a model for human motion behavior in the specific task of passing and slipping on a shoe to someone else's foot. We learn the human provider motion using the rigid-body $\mathrm{GP}_{\tilde{B}}$ parametrized by axis-angle pseudo-vector and Euclidean vector.

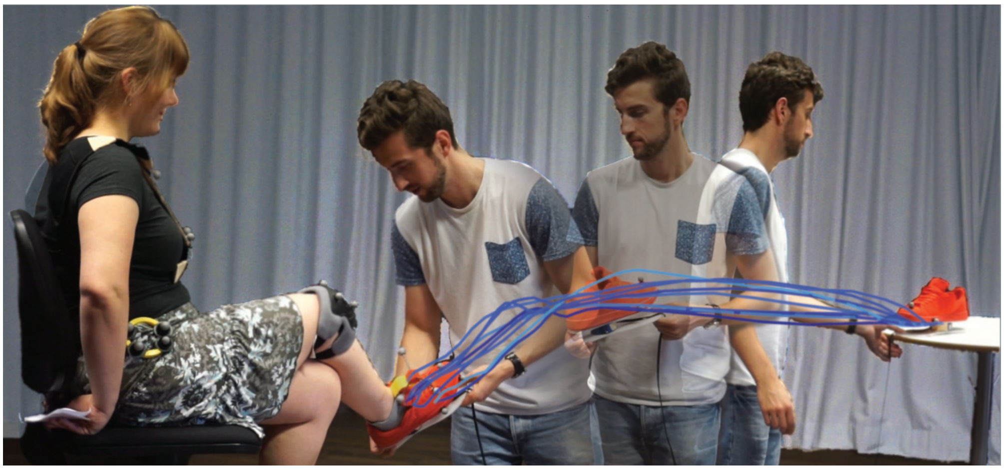
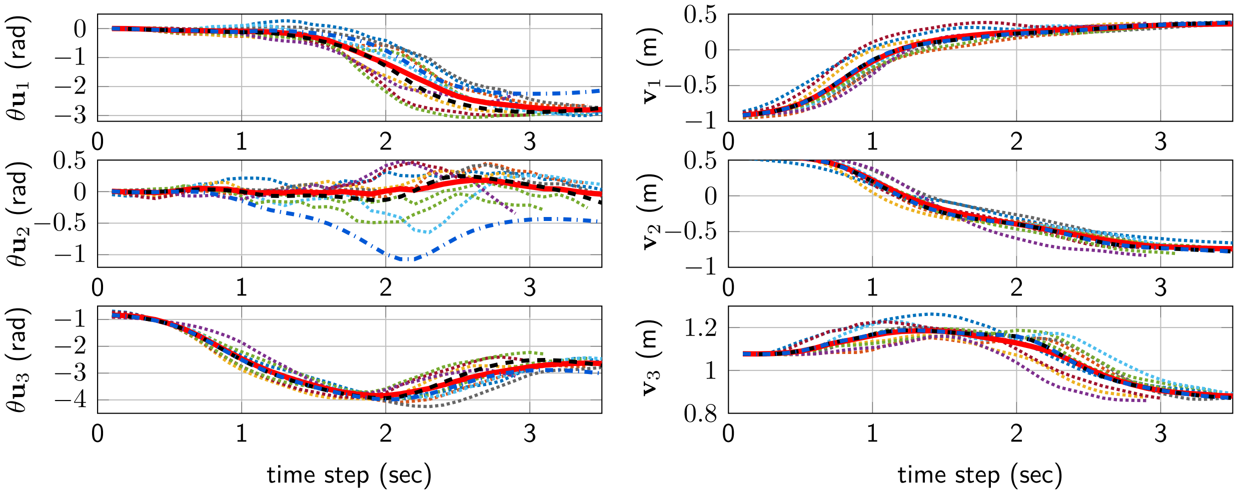


Figure 3.7: Human-human shoe providing experiment. One human puts on a shoe to another as preliminary study for elderly assistance in service robotics. The blue lines depict the translation part of the providers' 6D motion trajectories.



(a) Rotation dimensions (b) Translation dimensions

Figure 3.8: Visualization of the training data (colored dotted), the newly generated motion trajectory by GP prediction (dashed black), the mean of human demonstrations (red solid) and the standard GP prediction (blue dash-dotted).

Experiment Procedure

The setup for capturing GP training data is visualized in Figure 3.7. The translation part of the trajectory bundle is depicted in blue. The GP is trained on 9 trials of shoe passing and slipping on, demonstrated by different human providers. The data is captured with a motion tracking system (Qualisys, Sweden) at 179 Hz and consists of 6D poses (rotation and translation of the shoe while grasped by the human). The corresponding 6D rotational and translational velocities are obtained by finite differences. The motion trajectories are synchronized on the onset and cut when the contact force between foot and shoe exceeds a predefined threshold, as we suggest to change the controller at that moment. To assure fast runtime of the GP prediction, the GP training is performed on 10% random samples of the captured data. We set the weights in (3.38) to $\rho_1 = 0.7$ and $\rho_2 = 0.3$, to compensate for the magnitude difference in the captured rotations and translations. The GP training with the total number of $\nu = 1752$ training data pairs requires 8.05 seconds.

Results and Accuracy

After the training phase, we calculate 100 new motion trajectories, starting from previously unseen initial 6D poses. The trajectories are generated using single-step ahead mean prediction, which is integrated to pose by the exponential function. In Figure 3.8a and Figure 3.8b one predicted trajectory is visualized exemplary dimension wise by a black dashed line. The training trajectories are depicted by dotted lines for rotation and translation dimensions. The mean of the training trajectories is visualized by a red solid line and the dark blue dash-dotted line shows a trajectory, which starts at the same initial pose and is predicted using the traditional $\mathrm{GP}_{\mathbb{R}^6}$. All of the trajectories estimated using the $\mathrm{GP}_{\tilde{\mathcal{B}}}$ starting within the region of human starting poses, stayed throughout the movement within the variance of the human training data. The full trajectory prediction requires 0.84 seconds on average.

For accuracy evaluation we resample the training all to the same length of 3.5 seconds.

Then, we determine per time step mean and variance of the training trajectories and use the so obtained time series distribution as our ground truth. For both GP variants we calculate per time step the Bhattacharyya distance to our reference distribution. On average, the distance to the $\text{GP}_{\tilde{B}}$ is 2.9 and to the classic $\text{GP}_{\mathbb{R}^6}$ using Euler angles it is 4.4. In addition, a t-test is conducted to compare the Bhattacharyya distances per GP condition. We consider per GP variant the distances per time step as random samples from a distribution describing the similarity of the GP predictions and our ground truth distribution. The t-test shows a significant higher similarity for the $\text{GP}_{\tilde{B}}$ than for the $\text{GP}_{\mathbb{R}^6}$; $t(68) = 4.1, p = 0$. For translation, see Figure 3.8b, no significant difference between the GP variants is visible, whereas the rotation, see Figure 3.8a, is considerably better captured by the rigid-body $\text{GP}_{\tilde{B}}$ over axis-angle and Euclidean vector.

3.7.4 Simulation: Comparison of Computational Efficiency and Prediction Accuracy

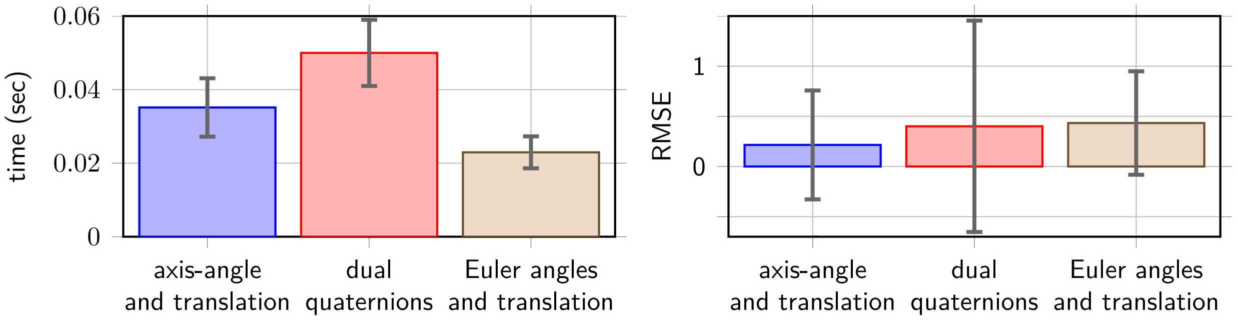
In this section we evaluate the generalized $\text{GP}_{\tilde{B}}$ over axis-angle and translation vector in two simulations; first regarding runtime of the GP prediction, and second regarding accuracy in the rotation estimation. We compare our results with the $\text{GP}_{\mathbb{H}_D}$ over dual quaternions and with the classic $\text{GP}_{\mathbb{R}^6}$ in Euclidean space, where rotations are represented inaccurately as Euler angles. We focus on the generalized squared exponential kernel (3.46), as for this kernel a pendant is available in quaternion space and as for the dot-product kernel no significant difference is expected, as calculations are equal to those in Euclidean space.

Runtime Comparison

We generate 80 training pairs $\mathcal{D}_\nu = \{(\mathfrak{g}_i,\ \dot{\mathfrak{g}}_{i,l})\}_{i=1}^{\nu}$ in this simulation, consisting of randomly drawn rigid-body motions $\mathfrak{g} \in SE(3)$ as GP input and corresponding velocity output $\dot{\mathfrak{g}}_l$ in one dimension $l \in \{1, \dots, 6\}$, which is obtained by the pseudo-random output generator algorithm provided in the GPML toolbox. Each training pair is once represented as axis-angle and translation vector, once as dual quaternion and once as Euler angles with translation vector. After the model training phase we compare the runtime required for GP velocity prediction at 10001 new randomly drawn rigid-body motions for the three representations. To obtain reliable and reproducible results, we repeat the experiment 1000 times. Figure 3.9a visualizes the resulting average prediction times of the 10001 test points and the standard deviation of the average required time for each representation. An independent-samples t-test is conducted to compare the runtime of the generalized rigid-body $\text{GP}_{\tilde{B}}$ with the $\text{GP}_{\mathbb{H}_D}$ over dual quaternions. The superiority of the $\text{GP}_{\tilde{B}}$ is verified by a t-test, $t(1998) = 37.8, p = 0$, showing a significant improvement in runtime.

Rotation Prediction Accuracy

In a second simulation we generate random rotation dynamics, mapping from the special Orthogonal group $SO(3)$ to the rotational velocities $TSO(3)$. We focus on rotations, as this is the well-known challenging issue in GP modeling. From every artificially generated dynamics f_{M_d} as in (2.5) we obtain 1000 ground truth testing pairs consisting each of a rotation with corresponding rotational velocity. From this data we draw 100 i.i.d. random training samples. After the GP training, we compare the predicted GP values for the testing



(a) Average GP velocity prediction runtime for 10001 unknown rigid-body motions $\mathfrak{g}$.

(b) RMSE in the GP prediction of 1000 test points for 1000 synthetically generated dynamics.

Figure 3.9: Runtime and prediction accuracy comparison. Per GP variant the box height shows the average required time and the average root-mean-square error (RMSE), respectively. The gray error bar depict the standard deviation over the 1000 experiment repetitions.

rotations with the ground truth velocities to evaluate the prediction accuracy for three rotational GP variants; the $\mathrm{GP}_{\tilde{B}}$ over axis-angle, the $\mathrm{GP}_{\mathbb{H}_D}$ over unit quaternions (which is a special case of the GP over dual quaternions with purely rotational input) and the $\mathrm{GP}_{\mathbb{R}^6}$ over Euler angles.

Procedure As we are interested in nonlinear motion behavior, we generate each rotation dynamics f_{M_d} randomly from the trigonometric functions $\mathcal{F} = \{\sin, \cos, \operatorname{atan}, \operatorname{acot}\}$. For simplicity of the description we consider vector-valued dynamics drift terms $f_{M_\mathsf{d}}(\theta\mathbf{u}) = \theta'\mathbf{u}'$ of a specific structure only: For $i = 1, \dots, 3$ each function component has the form

$$f_{M_\mathsf{d}}^{(i)}(\theta\mathbf{u}) = g\left(\frac{d}{c}\right) \diamond \tilde{d} \star \tilde{c}, \tag{3.77}$$

where function $g \in \mathcal{F}$. The operators are defined by $\diamond := \pm$ and $\star := \times/\div$, the constants $c, \tilde{c} \in \{\pm 1, \pm\frac{2}{3}, \pm\frac{1}{3}\}$ influence the rotation magnitude in the dynamics and the variables $d, \tilde{d} \in \{\theta\mathbf{u}_1, \theta\mathbf{u}_2, \theta\mathbf{u}_3\}$ denote entries in the axis-angle pseudo-vector. We randomly generate 1000 rotation dynamics of the described form, transform the axis-angle representation to unit quaternions and Euler angles for both, dynamics input and output and unwrap the Euler angles to avoid the jump at $\pm\pi$. Per simulated dynamics we train the models using the standard procedure of minimizing the negative log marginal likelihood, where the hyperparameter start values λ, σ_f are i.i.d. samples from the interval $(0, 3)$. The signal noise σ_n is set to 1 to assure numerical stability during the training phase. As evaluation, the single-step prediction accuracy of each test value in terms of the RMSE is compared to the ground truth.

Results It is clearly visible in Figure 3.9b that the average RMSE over 1000 repetitions of each time 1000 predictions of the generalized $\mathrm{GP}_{\tilde{B}}$ over axis-angle outperforms the other GP variants, as its estimation error is only about half the error of the other models. Independent-sampled t-tests are conducted to compare the axis-angle generalization with each of the other

GPs. A significant improvement (with significance level 1%) in the scores for both comparisons is obtained: $t(5998) = 8.6$, $p = 0$ for condition axis-angle versus dual quaternions and $t(5998) = 16$, $p = 0$ for condition axis-angle versus Euler angles. These results demonstrate the robustness of the $\mathrm{GP}_{\tilde{B}}$ over axis-angle, as it succeeds best in approximating the dynamics under such generalized and suboptimal learning circumstances. Due to the projection required in the $\mathrm{GP}_{\mathbb{H}_D}$ over unit and dual quaternions, respectively, this variant is highly sensitive to the starting values of the GP model learning and thus, performs inferior in learning a suitable GP model for many of the 1000 random dynamics. Hence, the results show large variance in the prediction error for the $\mathrm{GP}_{\mathbb{H}_D}$, even though theoretically comparable estimation quality should be achievable.

3.8 Discussion

To conclude this chapter, we provide a short summary of the characteristics of the developed GP generalization to $SE(3)$.

The generalized GP is defined on non-Euclidean input space and captures dynamics on the manifold $SE(3)$ mathematically firm. We present two well suited rigid motion parametrizations, namely the axis-angle and Euclidean vector and the dual quaternions. Additionally, general information for employing the homogeneous transformation matrix as rigid motion parametrization is provided.

For both investigated representations of the special Euclidean group $SE(3)$, we introduce a set of fundamental functions containing the zero mean and a dot-product kernel as well as a stationary kernel function and prove them to be valid. In a comparison of the presented GP generalization variants, we assess their optimal application scope: The generalized GP employing the axis-angle representation is computationally more efficient and accurate, but less suitable for approximating wide spread uncertainty on the manifold. The generalized GP employing dual quaternions is more vulnerable to bad training behavior depending on the initial choice of hyperparameters, but allows for the definition of a probability distribution on the full space of rotations.

Both generalized GPs outperform the state of the art $\mathrm{GP}_{\mathbb{R}^6}$ where rotations are represented by Euler angles. We evaluate the generalized version of the most widely used kernel, the squared exponential kernel, under various aspects such as computational complexity, prediction accuracy on simulated data. We analyze how the ground truth estimation accuracy of the $\mathrm{GP}_{\mathbb{H}_D}$ outperforms the classic $\mathrm{GP}_{\mathbb{R}^6}$ on estimating nonlinear dynamics. Further, the suitability of the GP generalizations for modeling real data involving freely performed human motions is proven experimentally. We contrast the characteristics of generalized GP kernels resulting from different metrics in 6D space in a real data experiment using human gaze motion. In this experiment we employ the dual quaternion representation. Additionally, the applicability of the generalized GP is assessed in a real world experiment involving natural human motion in the complex task of putting on a shoe. The generalized $\mathrm{GP}_{\tilde{B}}$ using the axis-angle and translation vector representation shows outstanding performance in terms of accuracy, unsusceptibility and computational efficiency. Hence, both generalized GP variants turn out to be suitable for approximating rigid motions, especially human motor behavior.

4

Human Decision Process Estimation through Parallel Planning and Weighting

In this chapter we present the methodology for our approximation of the human decision process dynamics (2.10). We start by reviewing related work and open problems concerning online movement generation including uncertainty and probabilistic trajectory classification.

Recall, we seek to approximate the potential movement plans

$$\left\{ \mathfrak{g}_{\mathsf{d}}^{1}(t) \right\}_{t\geq 0}, \dots, \left\{ \mathfrak{g}_{\mathsf{d}}^{\hat{\varrho}}(t) \right\}_{t\geq 0} \tag{4.1}$$

by a set of trajectory bundles $\dot{\cup}_{j=1}^{\hat{\varrho}} \mathcal{C}_j$, which contains in every bundle $\mathcal{C}_j$ multiple executions of assumedly each time the same potential desired trajectory $\{ \mathfrak{g}_{\mathsf{d}}^{j}(t) \}_{t\geq 0}$. Therefore, each trajectory

$$\mathfrak{x}_k = \left\{ \mathfrak{g}^{k}(t) \right\}_{t\geq 0} \tag{4.2}$$

in a set of captured human movement trajectories $\widetilde{\mathcal{D}}_\eta = \{ \mathfrak{x}_k \}_{k=1}^{\eta}$, needs to be assigned into the clusters representing the potential movement plans. Note, the exponent k in the trajectory formula (4.2) is an index denoting to which trajectory $\mathfrak{x}_k$ the rigid motion $\mathfrak{g}$ belongs.

Remark 4.0.1 *The set of trajectory bundles jointly represent the variety of potential desired trajectories, whereas each trajectory bundle on its own represents a potential desires trajectory, including the variance of human motion execution.*

Since we solely aim for application scenarios, where the potential movement plans terminate in distinct configurations in $SE(3)$, it suffices to cluster the trajectory end poses, to categorize the whole trajectory. Therefore, we review various data clustering algorithms and present our choice for clustering rigid motion data samples $\mathfrak{g} \in SE(3)$ in this chapter.

Subsequently, we present our approach for the potential desired trajectory weighting. Recall, the mapping (2.9) is used for calculating the probability vector $\mathbf{p}_{\mathfrak{g}} = (p_1, \dots, p_{\hat{\varrho}})^\top$ for a novel unseen rigid motion $\mathfrak{g}$ to belong to the clusters $\mathcal{C}_j$, $j = 1, \dots, \hat{\varrho}$. Now, we present how the mapping (2.9) is obtained explicitly. Therefore, we develop a distance measure on rigid motions $\mathfrak{g} \in SE(3)$, which exploits the properties of the Mahalanobis distance.

Then, a recursive algorithm of the generalized GP is developed to approximate the dynamics (2.10). Recall, we approximate the human internal decision making for a certain desired trajectory by incorporating the weight vector $\mathbf{p}_{\mathfrak{g}}$ into our approximation of the decision process dynamics $\tilde{f}_D$. Specifically, the estimated probabilities $p_1, \dots, p_{\hat{\varrho}}$ are used for *leveraging* the training data sample of the online generalized GP. This means, the samples are weighted according to the significance of the respective trajectory bundles $\mathcal{C}_1, \ldots, \mathcal{C}_{\hat{\varrho}}$.

Finally, we evaluate our approximation of the human decision process along a complex real world experiment on putting on shoes to another person and conclude the chapter with a brief summary of the decision process approximation.

4.1 Related Work and Open Problems

The relevant related work for the decision process approximation is two-fold: Since we aim to simultaneously represented potential spatial movement plans to possibly multiple target locations, an appropriate approach is required to generate the movement plans online. As we rely on human movements to approximate the potential movement plans, we have to consider uncertainty. According to [59] human motor execution includes sensorimotor uncertainty. Therefore, we review in the following approaches for generating movement plans online including uncertainty. Additionally, we require an algorithm for the decision making for a specific motion plan. Since we aim to approximate the human potential movement weighting, we again have to consider uncertainty and online capability. Therefore, we review in addition related work on probabilistic online trajectory classification.

Online Movement Generation including Uncertainty

A motion trajectory is the path, an object is following in space as a function of time. Very early approaches for trajectory planning in 6D space are available since 1998 already, see [97]. Among the many online trajectory planning algorithms, such as the probabilistic roadmap method (PRM) [98] or the rapidly-exploring random trees (RRTs) [99], we are interested in motion trajectory generators with confidence consideration, as the confidence is required to model the uncertainty in the human movement. Typically, the trajectory planners including confidence employ underlying Gaussian systems. We have reviewed predictive models for motion generation in the previous Chapter 3 already. So far, there is no mathematical firm model available for motion generation in rigid motion space based on Gaussian systems.

Since we require a method which is capable of generating the motion online, we present the Kernel Recursive Least-Squares (KRLS) algorithm introduced by [100]. This approach provides an online model for motion prediction using GPs. It remains an open problem, however, to generalize the KRLS algorithm to rigid motions. To take environmental information into account for in the trajectory planning using GPs, Choi et al. [101] present a method for weighting the sample reliability in the GP model learning. So far, however, the sample reliability has not been incorporated into the online movement generation considering uncertainty.

Probabilistic Trajectory Classification

Classification in general tries to find a rule to assign objects to predefined classes. Similarly, trajectory classification aims at determining the class label of trajectories from a predefined set of labels based on the characteristics of the trajectories. Besides data clustering, it is the main component of trajectory data mining [102]. Data clustering groups the data into a finite set of categories, also called clusters, based on their movement characteristics. Since we will introduce the most frequently used clustering algorithms more in detail in Section 4.2.1, we focus here on trajectory classification. Most trajectory classification algorithms follow a traditional two-step approach: first extracting a set of discriminative features (such as average speed or acceleration of a trajectory, trajectory duration, etc.) and then using the extracted features to train an existing standard classification model [102], such as support vector machines (SVMs) [103] or the decision tree algorithm [104]. Besides, the state of the art methods typically are based on hidden Markov models (HMMs) [38, 105], a classification approach that requires preceding trajectory segmentation. A shortcoming of the methods presented so far is that the can only provide a delayed class nomination.

An interesting line of research is proposed by Lee et al. [106] in the trajectory clustering framework *TraClass*, which partitions the trajectories and generates a hierarchy of features. The salient performance of the framework is due to exploring region-based and trajectory-based clustering in parallel. In [107] the approach was generalized from purely spatial to spatial or temporal trajectories. A generalization to trajectories in rigid motion space, consisting of rotation and translation, still is lacking.

In 2011 De Schutter [108] presented an approach to classify 6-DoF rigid motion trajectories using the invariant representation as input. A time-based coordinate-free description consisting of six scalar functions of time is defined based on a motion model for the instantaneous screw axis [109]. For the so represented motion trajectories two alternative approaches were introduced; one based on a Dynamic Time Warping algorithm and one based on Hidden Markov Models.

Since none of the trajectory classification methods is capable of labeling the class of a novel unseen trajectory online, we focus on point-wise classification of rigid motions instead of full motion trajectories. Additionally, none of the presented trajectory classification methods considers uncertainty. To approximate human movement plan weighing, however, a probabilistic method is required. To the best of the author's knowledge, there is no method available so far that allows for online trajectory classification in the special Euclidean group $SE(3)$ including uncertainty.

The most promising approach in that area is a probabilistic method based on maximum likelihood estimation [105]. As the required computational complexity of this approach is very limited, it is appropriate for online applications. Therefore, we suggest to exploit this line of research: the state of the art maximum likelihood estimation in online rigid motion trajectory classification.

4.2 Differentiation of Potential Movement Plans

In this section, we present our approach for rigid motion clustering. We start with providing an overview over major data clustering algorithms, stating their pros and cons. This induces the identification of the Expectation-Maximization (EM) algorithm using GMMs to match best our requirements for executed human movement data. Then, we introduce a generalization of the algorithm to allow for input on the rigid motion manifold $SE(3)$.

4.2.1 Data Specifications and Clustering Algorithms

We provide a set of human movement properties and infer a wishlist of cluster characteristics from them.

- The human sensorimotor uncertainty depends on the posture of the limbs and tension in the limbs. This means, data samples of human performed rigid motions have variable uncertainty, depending for example on whether the human had the limb stretched out or angled. Therefore, we require the algorithm, which is used to cluster the rigid motions, to allow for variable data densities and heterogeneous cluster sizes.
- We assume all intended human movements to be equally important, and therefore, consider the generation of an "outlier cluster" as undesired.
- Modeling human behavior using Gaussian based approaches has proven advantageous in many applications [110–113]. Therefore, we desire the clusters to be modeled by a (multivariate) normal distribution.
- We would like the algorithm to determine the optimal number of clusters, as this is not always intuitively obvious for unconstrained human behavior.

In the following we informally explain the main characteristics of the five most widely used clustering algorithm, which jointly represent the state of the art in data clustering, and subsequently justify our choice for the GMM clustering algorithm.

K-Mean Clustering

The probably most popular algorithm is the K-Means clustering [114]. The algorithm iteratively assigns the samples in the data set to a pre-selected set of "group centers" and updates each group center to mean of the assigned data subset. An advantage of this algorithm is its simplicity and computational efficiency. A disadvantage arises from the need to select the number of clusters in advance and the non-reproducability of the results, as the algorithm typically is initialized by randomly drawn initial "group centers".

Mean-Shift Clustering

This algorithm is based on a sliding-window approach. It searches for areas among the data points with high density. Therefore, it shifts a window iteratively to a region with higher density through moving its center to the mean of the data samples within the window. The shifting is continued until convergence. Then, new windows are introduced, one after the other, until all data is assigned to an area [115]. In contrast to K-means clustering this algorithm automatically detects the required number of clusters. A drawback, however, is that the size of the areas/windows has to be pre-selected.

Density-Based Spatial Clustering of Applications with Noise (DBSCAN)

One of the most cited algorithms is the DBSCAN. It was only recently introduced by [116] in 1996. It searches for dense data neighborhoods by assigning all samples to the same neighborhood that are "close" (i.e. the distance is below $\varepsilon > 0$) to an already assigned sample of the neighborhood. A new neighborhood can only be initialized, if it contains "enough" (i.e. more than $\nu_{\min} \in \mathbb{N}^+$) samples. All remaining samples are considered noise. This algorithm is good in finding almost arbitrarily shaped and sized clusters. However, it performs poorly on data sets with variable cluster densities, as the parameters ε and $\nu_{\min}$ are set in advance.

Hierarchical Agglomerative Clustering (HAC)

Initially each of the data samples is regarded as a separate cluster. Iteratively, the two most similar clusters (according to some arbitrary similarity measure, such as *single linkage* [117]) are merged into one cluster [118]. The algorithm ends when there is only one cluster left. An advantage is that this algorithm is not sensitive to the choice of distance metric. The shortcomings, however, include the computational inefficiency and the need to pre-select the number of clusters, in case the algorithm should be interrupted earlier.

GMM Clustering using the Expectation–Maximization (EM) algorithm

This algorithm requires pre-selection of the number of clusters, just like the K-mean algorithm. The cluster initialization, however, is not as critical, because the EM-algorithm is handling the cluster optimization. Iteratively the probability of each data point to belong to a particular cluster is calculated, and the parameters of the Gaussian distribution are optimized to best describe the weighted data points [66, Chap. 9.2.2]. Since the algorithm uses Gaussians as clusters, their shape is described by two parameters, the mean and the covariance, which define an efficient way of describing the less restrictive shape of an ellipsoid. Additionally, as the GMM clustering employs Gaussian probabilities, a single data sample can be assigned to multiple clusters, i.e. this algorithm allows for mixed memberships.

Even though the requested pre-selection of the number of clusters is a disadvantage, we suggest to use this algorithm for human motor behavior clustering: The clusters are defined by Gaussians, that are well suited for modeling human movements and allow for variable cluster size and density. Additionally, outlier clusters are not supported, and the algorithm is computationally efficient. To handle the pre-selection of the cluster size, the algorithm is run multiple times for different cluster sizes and determine afterwards the optimal number according to the Bayesian information criterion (BIC) [119]. We introduce this clustering algorithm more in depth in the following section.

4.2.2 Generalization of GMM Clustering to Rigid Motions

In this section we introduce a GMM cluster generalization to data samples on the manifold $SE(3)$. The fitting of the generalized Gaussians to the data set is performed using a modified EM algorithm, which initially was introduced already in 1977 in [120]. Its underlying idea is to fit Gaussian distributions to a data set by alternating the steps to stochastically assign the data to the clusters and to maximize the likelihood of the Gaussians to describe those clusters. In the generalization, we need to adapt the probability distribution to be defined in $SE(3)$ and we require a distance function over rigid motion $\mathfrak{g} \in SE(3)$. We propose to use the projected Gaussian distribution, introduced in Section 3.5.1 and the dual quaternion distance metric (3.35).

An alternative approach, investigating pure rotation parametrized by unit quaternions, was introduced in 2017 by [121]. The authors suggest to use an exponential map to transform from the rotations into the velocity space. However, the mappings $\exp(\cdot)$ respectively $\ln(\cdot)$ only allow for defining the covariance in a local neighborhood around the tangent point in rotation space (see Section 3.5.2). Therefore, we use the central projection (3.56) in contrast to map rigid motions on the manifold into the velocity space, as the projection $\Pi_{\mathfrak{g}}$ allows for defining a projected Gaussian distribution on the whole rigid motions.

Remark 4.2.1 *Since switching the rigid motion representation is simple using* (3.19)*, we will not distinguish whether our training data set* $\mathcal{D}_\nu$ *is represented by dual quaternions or by pairs of a unit quaternion and a translation vector, i.e.* $\mathfrak{g} = (\,\mathbf{q},\, \mathbf{v})$*. While we introduced comprehensive calculus for the dual quaternion parametrization in Chapter 3, algebraic calculation rules for the representation as a pair are well-known.*

The generalized algorithm itself comprises four steps:

Initalization

The only necessary parameters to fully specify a generalized GMM with $\hat{\varrho}$ components (i.e. cluster size $\hat{\varrho}$) are the mixing coefficients χ_j, $j = 1, \dots, \hat{\varrho}$ and the projected Gaussian distribution parameters μ_j, Σ_j at their corresponding tangent points $\mathfrak{g}_j$ for $j = 1, \dots, \hat{\varrho}$. The factors χ_j define a convex combination $\sum_{j=1}^{\hat{\varrho}} \chi_j = 1$.

To avoid coincidental poor optimization performance in the subsequent iterations, the mean values are not sampled purely at random. Specifically, we sample at random mean values $\mu_1, \dots, \mu_{\hat{\varrho}}$ without replacement, which fulfill the condition

$$\sum_{i=1}^{\nu} d_{\text{mag}}(\mu_j,\, \mathfrak{g}_i) \leq c_{\max}, \tag{4.3}$$

for some pre-selected upper bound $c_{\max} \in \mathbb{R}^+$. In case a mean value does not meet (4.3), we discard it and re-sample. The first value μ_1 is sampled from a uniform distribution. From then on, the following means μ_j for $j = 2, \dots, \hat{\varrho}$ are sampled from weighted training data

$$\check{\mathcal{D}}_{\nu,j} = \left\{ \frac{d_{\text{mag}}(\mu_{j-1},\, \mathfrak{g}_i)}{\sum_{i=1}^{\nu} d_{\text{mag}}(\mu_{j-1},\, \mathfrak{g}_i)}\, \mathfrak{g}_i \right\}_{i=1}^{\nu} \tag{4.4}$$

to facilitate a homogeneous distribution of the means. The mixing coefficients are initialized all with the same value

$$\chi_j = 1/\hat{\varrho} \text{ for all } j \in \{1, \dots, \hat{\varrho}\}, \tag{4.5}$$

and the initial covariance matrices Σ_j for all $j = 1 \dots, \hat{\varrho}$ are calculated as the stacked diagonal entries of the pure rotational and pure translational variance of the whole training data set $\mathcal{D}_\nu$

$$\Sigma_j = (\operatorname{diag}(\Sigma_{\text{rot}}), \operatorname{diag}(\Sigma_{\text{tra}}))^\top. \tag{4.6}$$

The calculation of Σ_{tra} is trivial using the formula $\mathbb{E}\left[(\{\,\mathfrak{g}_i\}_{i=1}^{\nu} - \mu_{\text{tra}})^2\right]$, where the translation mean $\mu_{\text{tra}} = 1/\nu \sum_{i=1}^{\nu} \mathfrak{g}_i$. Analogously, the mean over unit quaternions μ_{rot} is calculated in compliance with [122] as the eigenvector, which corresponds to the largest eigenvalue of the auto-correlation matrix

$$\mathbf{M}_{\text{a:corr}} = \sum_{i=1}^{\nu} \mathbf{q}_i\, \mathbf{q}_i^\top. \tag{4.7}$$

Then, the rotational variance Σ_{rot} is obtained in the tangent space to μ_{rot} as the standard Euclidean variance of the pullback of the complete rotation data set $\{\,\mathbf{q}_i\}_{i=1}^{\nu}$ into the tangent space of μ_{rot} using (3.53),

$$\Sigma_{\text{rot}} = \mathbb{E}\left[\left(\Pi^{-1}_{\mu_{\text{rot}}}(\mu_{\text{rot}}\{\,\mathfrak{g}_i\}_{i=1}^{\nu}) - \mu_{\text{rot}}\right)^2\right]. \tag{4.8}$$

E-step

Given the current parameter values, we calculate the responsibilities of the data set $\mathcal{D}_\nu$. This means, for each sample $\mathfrak{g}_i \in \mathcal{D}_\nu$ for $i = 1, \dots, \nu$ a vector $(\rho_1, \dots, \rho_{\hat{\varrho}})$ is computed, encoding the probabilities of $\mathfrak{g}_i$ to belong to the clusters $\mathcal{C}_1, \dots, \mathcal{C}_{\hat{\varrho}}$.

A $\nu \times \hat{\varrho}$ responsibility matrix $\mathbf{P}$ (where the matrix entries $(\mathbf{P})_{i,j} = \rho_{ij}$) encoding the probabilities per sample and per cluster is obtained by evaluating

$$\rho_{ij} = \frac{\chi_j \, \mathcal{N}\left(\mathfrak{g}_i \middle| \, \mathfrak{g}_j, \mu_j, \Sigma_j \right)}{\sum_{k=1}^{\hat{\varrho}} \chi_k \, \mathcal{N}\left(\mathfrak{g}_i \middle| \, \mathfrak{g}_k, \mu_k, \Sigma_k \right)} \,. \tag{4.9}$$

The projected Gaussian distribution $\mathcal{N}\left(\mathfrak{g}_j, \mu_j, \Sigma_j \right)$, as defined in Section 3.5.1 applying the projection (3.56), inherits the Gaussian distribution properties, but is defined on the manifold $SE(3)$. To obtain the individual probabilities $\mathcal{N}\left(\mathfrak{g}_i \middle| \, \mathfrak{g}_j, \mu_j, \Sigma_j \right)$ for a given sample $\mathfrak{g}_i$ and cluster $\mathcal{C}_j$ the projected Gaussian probability density function (3.54) is employed.

M-step

Given the current responsibility matrix $\mathbf{P}$, we update the defining generalized GMM parameters $\tilde{\chi}_j, \tilde{\mu}_j$ and $\tilde{\Sigma}_j$ for each cluster $\mathcal{C}_j$, $j = 1, \dots, \hat{\varrho}$.

The updated mean $\tilde{\mu}_j$ of the j-th cluster $\mathcal{C}_j$ is the weighted average of the samples $\mathfrak{g}_i$ with $i = 1, \dots, \nu$. We separately calculate the translation and rotation components of the mean and concatenate it afterwards into one vector. Applying the weight vector $(\mathbf{P})_{\cdot,j}$ of the responsibility matrix, we obtain for the translation part $\mathbf{v}_i$ in $\mathfrak{g}_i$ the new mean

$$\tilde{\mu}_{\mathrm{tra}_j} = \frac{\sum_{i=1}^{\nu} \rho_{ij} \, \mathbf{v}_i}{\sum_{i=1}^{\nu} \rho_{ij}}. \tag{4.10}$$

For the rotation part $\mathbf{q}_i$ in $\mathfrak{g}_i$, we employ Markley's [122] unit quaternion averaging algorithm: We compute the weighted auto-correlation matrix

$$\mathbf{M}_{\mathrm{a:corr}} = \sum_{i=1}^{\nu} \rho_i \, \mathbf{q}_i \, \mathbf{q}_i^\top \tag{4.11}$$

of the full sample set and obtain the updated rotation mean $\tilde{\mu}_{\mathrm{rot}_j}$ as the eigenvector, which corresponds to the largest eigenvalue of matrix (4.11). The complete new mean is then given by $\tilde{\mu}_j = (\tilde{\mu}_{\mathrm{tra}_j}^\top, \tilde{\mu}_{\mathrm{rot}_j}^\top)^\top$.

The updated covariance $\tilde{\Sigma}_j$ for $j = 1, \dots, \hat{\varrho}$ is given by

$$\tilde{\Sigma}_j = \frac{1}{\sum_{i=1}^{\nu} \rho_{ij}} \sum_{i=1}^{\nu} \rho_{ij} \, \mathbf{v}_{\Delta TS} \, \mathbf{v}_{\Delta TS}^\top, \tag{4.12}$$

where the delta vector $\mathbf{v}_{\Delta TS}$ is a concatenation of the delta translations from $\tilde{\mu}_{\mathrm{tra}_j}$ to $\mathbf{v}_i$ and the pullback into the velocity space using $\Pi_{\mu_{\mathrm{rot}_j}}^{-1}$ of the delta rotation from μ_{rot_j} to $\mathfrak{g}_i$

$$\mathbf{v}_{\Delta TS} = \left[\left(\mathbf{v}_i - \tilde{\mu}_{\mathrm{tra}_j} \right)^\top, \left(\Pi_{\mu_{\mathrm{rot}_j}}^{-1} \left(\mu_{\mathrm{rot}_j} \, \mathfrak{g}_i \right) \right)^\top \right]^\top . \tag{4.13}$$

Note, the notation $\mathbf{v}_{\Delta TS}$ in this case represents a spatial difference between mean and the data samples, no temporal difference. Finally, the updated mixing coefficients $\tilde{\chi}_j$ for $j = 1, \dots, \hat{\varrho}$ are given by

$$\tilde{\chi}_j = \frac{1}{\nu} \sum_{i=1}^{\nu} \rho_{ij}. \tag{4.14}$$

Hence, we have updated all characteristic parameters and a new generalized GMM is defined by $\tilde{\mu}_j, \widetilde{\Sigma}_j$ and $\tilde{\chi}_j$ for $j = 1, \dots, \hat{\varrho}$.

Likelihood

To check for convergence of the EM algorithm, we evaluate the log likelihood of the fitted generalized GMM. In the **M-step** we computed the Gaussian parameters μ_j, Σ_j in the tangent space $T_{\mathfrak{g}_j} SE(3)$ to the rigid motions and have not yet projected the Gaussian distributions into the manifold using (3.56) (this is done in the **E-step**). Therefore, we can simply employ the standard log likelihood formula

$$\ln\left(p\left(\{\mathfrak{g}_i\}_{i=1}^{\nu} \middle| \boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\chi} \right) \right) = \sum_{i=1}^{\nu} \left(\sum_{j=1}^{\hat{\varrho}} \chi_j \mathcal{N}(\mathfrak{g}_i | \mu_j, \Sigma_j) \right). \tag{4.15}$$

In case (4.15) does not satisfy the convergence criterion, namely $\ln p(\{\mathfrak{g}_i\}_{i=1}^{\nu} \big| \boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\chi})$ has increased about more than $\varepsilon > 0$ in comparison to the previous iteration, we return to the **E-step** and proceed again from there on.

After the algorithm has converged, we calculate the BIC value, which is a criterion for describing the quality of the model fit,

$$\mathrm{BIC} = \underbrace{\frac{\dim(\mathfrak{g}) + 1}{2} \hat{\varrho} \dim(\mathfrak{g}) \ln(\nu)}_{\text{penalty term}} - 2 \ln\left(p\left(\{\mathfrak{g}_i\}_{i=1}^{\nu} \middle| \boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\chi} \right) \right). \tag{4.16}$$

The lower the BIC, the better the fit. Since in GMM modeling the likelihood is increased by adding clusters (until every data sample has its own cluster), which leads to overfitting, the BIC criterion includes a penalty term on the number of parameters in the model.

4.3 Probabilistic Classification of Desired Trajectory

In this section we present our technique to weight and classify movement trajectories. Given bundles of movement trajectories (the clusters $\mathcal{C}_1, \dots, \mathcal{C}_{\hat{\varrho}}$, which we assigned using the generalized GMM clustering introduced in Section 4.2.2), we estimate *over the course of the ongoing movement* the probability of an unseen rigid motion trajectory

$$\mathfrak{x}_{\eta+1} = \left\{ \mathfrak{g}^{\eta+1}(t) \right\}_{t \geq 0} \subset SE(3) \tag{4.17}$$

to belong to one of the clusters. Recall, the superscript $\eta + 1$ denotes the trajectory, the rigid motion $\mathfrak{g}$ belongs. We also consider the option that the novel trajectory belongs to none of the clusters, but defines a new cluster $\mathcal{C}_{\hat{\varrho}+1}$.

Procedure

We desire the decision process to operate online. Therefore, we compute throughout the ongoing movement trajectory $\mathfrak{x}_{\eta+1}$ a probability vector $\mathbf{p}_{\mathfrak{g}^{\eta+1}(t)} = (p_1^{\eta+1}(t), \ldots, p_{\hat{\varrho}}^{\eta+1}(t))^\top$ for any rigid motion $\mathfrak{g}^{\eta+1}(t)$. The probability vector $\mathbf{p}_{\mathfrak{g}^{\eta+1}(t)}$ is obtained from a similarity measure between the ongoing movement and the given trajectory bundles $\mathcal{C}_1, \ldots, \mathcal{C}_{\hat{\varrho}}$. This similarity measure defines the mapping (2.9) which we introduced in Section 2.2. In the following section, we present how (2.9) is calculated.

Afterwards, we introduce a set of decision rules to support the assignment of the trajectory $\mathfrak{x}_{\eta+1}$ to the corresponding desired trajectory bundle $\left\{ \mathfrak{g}_{\mathsf{d}}^{j}(t) \right\}_{t \geq 0}$ for $j \in \{1, \ldots, \hat{\varrho}\}$, depending on the probabilities $\mathbf{p}_{\mathfrak{g}^{\eta+1}(t)}$.

4.3.1 Similarity Measure and Class Probabilities

As we consider the decision process dynamics (2.10) for any unseen movement, we need to introduce the mapping (2.9) to obtain the probabilities of the trajectory bundles $\uplus_{j=1}^{\varrho} \mathcal{C}_j$. The method, we propose therefore, follows the state of the art classification based on maximum likelihood estimation, as introduced in [105]:

To classify a new motion trajectory $\mathfrak{x}_{\eta+1} = \{ \mathfrak{g}^{\eta+1}(t)\}_{t \geq 0}$ into the set of known trajectory bundles $\uplus_{j=1}^{\varrho} \mathcal{C}_j$ we employ at any time $t \geq 0$ a class-conditional likelihood term

$$p(\mathfrak{g}^{\eta+1}(t) | \Theta_j, \mathcal{C}_j) \tag{4.18}$$

to estimate the cluster probabilities $(p_1^{\eta+1}(t), \ldots, p_{\varrho}^{\eta+1}(t))^\top$ for the rigid motion $\mathfrak{g}^{\eta+1}(t)$. The set of low level model parameter estimates Θ_j consists of the defining generalized GMM parameters of the j-th cluster

$$\Theta_j = \{\mu_j, \Sigma_j, \chi_j\} \text{ for } j = 1, \ldots, \varrho. \tag{4.19}$$

The similarity measure we use inside the class-conditional likelihood term is based on the Mahalanobis concept, but generalizes the Mahalanobis distance to the manifold $SE(3)$. More precisely, we provide a similarity measure for a newly observed trajectory $\mathfrak{x}_{\eta+1}$ and a set of training trajectories, $\widetilde{\mathcal{D}}_\eta = \{ \mathfrak{x}_k\}_{k=1}^{\eta}$, which is based on a normalized inverted Mahalanobis distance over rigid motions. Therefore, we introduce the generalized Mahalanobis distance on the manifold $SE(3)$:

Definition 4.3.1 *The generalized Mahalanobis distance of a new rigid motion $\mathfrak{g}_{\nu+1} \in SE(3)$ and a set of rigid motions $\{ \mathfrak{g}_1, \ldots, \mathfrak{g}_\nu\} \subset SE(3)$ is defined as*

$$d_{\mathrm{maha}}(\mathfrak{g}_{\nu+1}, \{ \mathfrak{g}_1, \ldots, \mathfrak{g}_\nu\}) = \sqrt{\begin{pmatrix} d_{\mathrm{m},1} & \ldots & d_{\mathrm{m},\nu} \end{pmatrix} \begin{pmatrix} k_{1,1} & \ldots & k_{1,\nu} \\ \vdots & \ddots & \vdots \\ k_{\nu,1} & \ldots & k_{\nu,\nu} \end{pmatrix}^{-1} \begin{pmatrix} d_{\mathrm{m},1} \\ \vdots \\ d_{\mathrm{m},\nu} \end{pmatrix}}, \tag{4.20}$$

where $d_{\mathrm{m},i}$ for $i = 1, \ldots, \nu$ either the dual quaternion distance measure $d_{\mathrm{mag}}(\mathfrak{g}_{\nu+1}, \mathfrak{g}_i)$ as introduced in (3.35) or the axis angle and translation vector distance measure $d_{\mathrm{wmag}}(\mathfrak{g}_{\nu+1}, \mathfrak{g}_i)$ as introduced in (3.38). Respective to the rigid motion representation in the distance function, the kernel functions $k_{i,j}$ are either defined by $k_{\mathrm{se:mag}}(\mathfrak{g}_i, \mathfrak{g}_j)$ as in (3.44) or by $k_{\mathrm{se:wmag}}(\mathfrak{g}_i, \mathfrak{g}_j)$ as in (3.46), with $i, j = 1, \ldots, \nu$.

Remark 4.3.1 *For the explanation of the similarity measure and the class-conditional likelihood, we require to know to which cluster* $\mathcal{C}_j$ *which of the training trajectories* $\mathfrak{x}_k = \left\{ \mathfrak{g}^k(t) \right\}_{t \geq 0}$*,* $k \in \{1 \ldots, \eta\}$ *are assigned. Therefore, we indicate in the reminder of this section the cluster membership by a subscript index in addition to the superscript index. Thus, variable* $\mathfrak{g}_j^k(t)$ *denotes the rigid motion at time* $t \geq 0$ *of the* k*-th execution in the* j*-th spatial motion plan.*

Class-conditional Likelihood for the Rigid Motion $\mathfrak{g}^{\eta+1}(t)$

To enable the decision process to operate online, we measure for any new trajectory $\mathfrak{x}_{\eta+1}$ the similarity against the training trajectories $\left\{ \mathfrak{g}_j^k(t) \right\}_{t \geq 0}$ in terms of *rigid motion similarity* across all conditions $j = 1, \ldots, \varrho$. Hence, we select for every trajectory $\mathfrak{x}^k$ in condition $\mathcal{C}_j$, the rigid motion $\mathfrak{g}_j^k(t')$ of an arbitrary time $t' \geq 0$, which is closest to the current rigid motion $\mathfrak{g}^{\eta+1}(t) \in \mathfrak{x}_{\eta+1}$ according to the distance measure (3.35) or (3.38), respectively (depending on the rigid motion parametrization). Then, we calculate the *inverse* d_{maha}^{-1} of the generalized Mahalanobis distance (4.20),

$$d_{\text{maha}}^{-1} = d_{\text{maha}}^{-1} \left(\mathfrak{g}_{\nu+1}(t), \left\{ \mathfrak{g}_j^k(t') \right\} \Big|_{k \in \{1, \ldots, \eta\}, \, \mathfrak{x}^k \in \mathcal{C}_j} \right), \tag{4.21}$$

to measure the *similarity* between the current rigid motion $\mathfrak{g}^{\eta+1}(t)$ of the new ongoing movement trajectory $\mathfrak{x}_{\eta+1}$ and the set of closest rigid motions $\left\{ \mathfrak{g}_j^k(t') \right\} \Big|_{k \in \{1, \ldots, \eta\}, \, \mathfrak{x}^k \in \mathcal{C}_j}$ of condition j, i.e. cluster index j is fixed, trajectory index $k \in \{1, \ldots, \eta\}$ restricted to $\mathfrak{x}^k \in \mathcal{C}_j$ is variable.

To obtain a probability $p(\mathfrak{g}_{\nu+1}(t) | \hat{\Theta}_j, \mathcal{C}_j)$ from the similarity measure for each cluster $\mathcal{C}_j$, the sum of the similarity measures across the clusters $\mathcal{C}_1, \ldots, \mathcal{C}_\varrho$ is normalized to 1,

$$p(\mathfrak{g}_{\nu+1}(t) | \hat{\Theta}_j, \mathcal{C}_j) = \frac{d_{\text{maha}}^{-1} \left(\mathfrak{g}_{\nu+1}(t), \left\{ \mathfrak{g}_j^k(t') \right\} \Big|_{k \in \{1, \ldots, \eta\}, \, \mathfrak{x}^k \in \mathcal{C}_j} \right)}{\sum_{i=1}^{\varrho} d_{\text{maha}}^{-1} \left(\mathfrak{g}_{\nu+1}(t), \left\{ \mathfrak{g}_i^k(t') \right\} \Big|_{k \in \{1, \ldots, \eta\}, \, \mathfrak{x}^k \in \mathcal{C}_i} \right)}. \tag{4.22}$$

The probabilities for all clusters $\mathcal{C}_1, \ldots, \mathcal{C}_\varrho$ taken together define the cluster weight vector $\mathbf{p}_{\mathfrak{g}_{\nu+1}(t)} = \left(p(\mathfrak{g}_{\nu+1}(t) | \hat{\Theta}_1, \mathcal{C}_1) \ldots, p(\mathfrak{g}_{\nu+1}(t) | \hat{\Theta}_\varrho, \mathcal{C}_\varrho) \right)^\top$ as introduced in (2.8).

Remark 4.3.2 *The equation* (4.22) *defines the mapping* (2.9)*, which we use to approximate the human internal movement weighting.*

Class-conditional Likelihood for the Trajectory $\mathfrak{x}_{\nu+1}$

From the class-conditional likelihood at $\mathfrak{g}^{\eta+1}(t)$, a class-conditional likelihood for the whole ongoing rigid motion $\mathfrak{x}_{\nu+1}$ is calculated by integrating the likelihood term $p(\mathfrak{g}_{\nu+1}(t) | \hat{\Theta}_j, \mathcal{C}_j)$ over time,

$$p(\mathfrak{x}_{\nu+1}(T) | \hat{\Theta}_j, \mathcal{C}_j) = \int_0^T p(\mathfrak{g}_{\nu+1}(t) | \hat{\Theta}_j, \mathcal{C}_j) dt, \tag{4.23}$$

where the time limit $T \geq 0$ encodes the upper bound of the continuous time interval $[0, T]$ and thus, defines the progress of the trajectory $\mathfrak{x}_{\nu+1}(T) = \{ \mathfrak{g}^{\eta+1}(t) \}_{t \in [0,T]}$.

Remark 4.3.3 *We require the potential movement weighting to operate online, probability based to approximate human decision making. The state on the art regarding online probabilistic classification method is based on the likelihood function. Therefore, we followed this approach and introduced a classification method for rigid motions based on the likelihood function. The presented class-conditional likelihood* (4.22) *is used to calculate the mapping* (2.9) *which we require for the movement weighting in the decision process dynamics* (2.10)*.*

4.3.2 Classification Rules

In the following, we introduce a set of decision rules to support the assignment of the trajectory $\mathfrak{x}_{\eta+1}$ to the corresponding desired trajectory bundle $\left\{\mathfrak{g}_{\mathfrak{d}}^{j}(t)\right\}_{t\geq 0}$ for $j \in \{1, \ldots, \hat{\varrho}\}$. We define four decision rules; two for nominating and two for eliminating classes.

On the one hand, the classification probability defined in (4.22) is contrasted against preset absolute thresholds to nominate or eliminate the candidate cluster when the probability falls outside of these thresholds. On the other hand, we implement a sliding window technique in parallel to either nominate or eliminate a potential spatial movement plan out of the set of trajectory bundles based on the magnitude of their relative probabilities. In this latter method, the integral of the probability within the most recent time window of length m is calculated using (4.23), i.e. we integrate $\int_{T-m}^{T} p(\mathfrak{g}_{\nu+1}(t)|\hat{\Theta}_j, \mathcal{C}_j)dt$ for every non-eliminated cluster $\mathcal{C}_j$. The window is then moved along the trajectory in real time. Therefore, this method initiates only when m time has passed after the trajectory onset.

When the probability allocation exceeds or undergoes predefined thresholds, the corresponding cluster $\mathcal{C}_j$, $j = 1, \ldots, \hat{\varrho}$ is nominated or eliminated from classification, respectively. The classification is terminated when a motion cluster is nominated. When the elimination criteria are met, the procedure resumes without the eliminated condition. Only the clusters that have not yet been eliminated have to be considered in the parallel motion prediction process of dynamics (2.10).

Remark 4.3.4 *For each time $t \geq 0$ in the dynamics* (2.10)*, when a certain trajectory bundle $\mathcal{C}_j$, $j \in \{1, \ldots, \hat{\varrho}\}$ can be eliminated, the desired motion prediction for this trajectory bundle is abandoned. Respectively, in case one trajectory bundle can be nominated at some time instance $t \geq 0$, only the motion prediction according to this trajectory bundle is pursued.*

4.4 Online Trajectory Generation including Sample Leveraging

In this section we present our approach to approximate the potential movement generation in the decision process. Previously, we discussed various GP models already that allow for rigid motion prediction. The models considered so far, however, are suitable for offline model learning only. Those models are convenient for applications with static environment. In our application domain, where we aim to approximate human decision making, the circumstances are time-varying, and hence, a-priori trained models are impractical, as they might be outdated, when it comes to prediction. Hence, the training data can be meaningless for the test case. Therefore, we introduce in the following a recursive algorithm that allows for online model training in the generalized GP framework. It is inspired by the Kernel

Recursive Least-Squares (KRLS) algorithm by [100] for standard GP models in Euclidean space. Additionally, we include the cluster probabilities in the algorithm through adapting the importance of the training data samples. We call this adaptation *sample leveraging.*

Let us consider a dynamical system of form $\dot{\mathfrak{g}} = \tilde{f}_D(\mathfrak{g}, \mathbf{p}_{\mathfrak{g}}, \mathbf{n}_{\mathfrak{g}})$ over rigid motions $\mathfrak{g}$ as introduced in (2.10), and a training data set $\mathcal{D}_\nu := \left\{(\mathfrak{g}_i,\, \dot{\mathfrak{g}}_i)\right\}_{i=1}^{\nu}$ consisting of ν input-output pairs from the dynamics $\tilde{f}_D = (\varphi_1, \dots, \varphi_n)^\top$, $n \in \mathbb{N}$ evolving on $SE(3)$. As each output component is processed separately and joined into one output distribution afterwards, see Sec. 3.2, we denote by $\dot{\mathfrak{g}}_i^{(l)}$ the l-th component of the output velocity for $l = 1, \dots, n$ and employ per GP the according training data set $\mathcal{D}_\nu^{(l)} := \left\{(\mathfrak{g}_i,\, \dot{\mathfrak{g}}_i^{(l)})\right\}_{i=1}^{\nu}$.

Then, the online GP algorithms comprises two main steps. First, it provides an n-dimensional predictive distribution for any new input $\mathfrak{g}_{\nu+1}$ given training data $\mathcal{D}_\nu$ and second, per component l it updates the joint Gaussian distribution

$$p\left(\varphi_l(\mathfrak{g}_1), \dots, \varphi_l(\mathfrak{g}_\nu)|\mathcal{D}_\nu^{(l)}\right) = \mathcal{N}\left(\varphi_l(\mathfrak{g}_1), \dots, \varphi_l(\mathfrak{g}_\nu)|\mu_\nu^{(l)}, \Sigma_\nu^{(l)}\right) \tag{4.24}$$

to a new conditional posterior $p(\varphi_l(\mathfrak{g}_1),\, \dots,\, \varphi_l(\mathfrak{g}_{\nu+1})|\mathcal{D}_{\nu+1}^{(l)})$, which includes the new observation $(\mathfrak{g}_{\nu+1},\, \dot{\mathfrak{g}}_{\nu+1}^{(l)})$. The mean vector $\mu_\nu^{(l)} \in \mathbb{R}^\nu$ and the covariance matrix $\Sigma_\nu^{(l)} \in \mathbb{R}^{\nu\times\nu}$ in (4.24) describe mean and covariance, respectively, of the prediction vector $(\varphi_l(\mathfrak{g}_1),\, \dots,\, \varphi_l(\mathfrak{g}_\nu))^\top$ in the l-th GP component for $l = 1, \dots, n$.

We employ leveraging of the training samples in each of the l online GPs, i.e. the samples are weighted according to the significance of the respective trajectory bundles $\mathcal{C}_1, \dots, \mathcal{C}_{\hat{\varrho}}$. We incorporate the weight of the training data into the kernel function following the leveraging method of [101].

In the reminder of this section we introduce the *generalized online leverage GP.* We start with the sample leveraging, where we explain, how the potential movement probabilities (which we calculated in Section 4.3.1) influence the GP prediction. Subsequently, we present the initialization and the update step of the recursive GP algorithm, and discuss characteristic properties.

4.4.1 Sample Leveraging

To employ the probability vectors $\mathbf{p}_{\mathfrak{g}(t)}$ in the motion prediction of dynamics (2.10), we leverage the training data samples according to the significance of the trajectory bundles. Specifically, the probability (4.22) of the j-th cluster is incorporated into a new kernel function for all rigid motions $\mathfrak{g}(t)$ that belong to the cluster $\mathcal{C}_j$. In the following, we enhance the generalized squared exponential kernels $k_{\text{se:mag}}(\mathfrak{g}_1,\, \mathfrak{g}_j)$ and $k_{\text{se:wmag}}(\mathfrak{g}_1,\, \mathfrak{g}_j)$, introduced in (3.44) and (3.46), respectively, to exploit the leverage of training samples.

Given the rigid-motion wise class-conditional likelihood, defined in (4.22), we obtain the cluster probability vector $\mathbf{p}_{\mathfrak{g}}$ at the rigid motion $\mathfrak{g}(t)$ as

$$\mathbf{p}_{\mathfrak{g}(t)} = \left(p(\mathfrak{g}(t)|\hat{\Theta}_1,\, \mathcal{C}_1) \dots, p(\mathfrak{g}(t)|\hat{\Theta}_{\hat{\varrho}},\, \mathcal{C}_{\hat{\varrho}})\right)^\top, \tag{4.25}$$

for the cluster set $\uplus_{j=1}^{\hat{\varrho}}\mathcal{C}_j$ and Θ_j the low level cluster parameters for $j = 1\dots, \hat{\varrho}$. The vector $\mathbf{p}_{\mathfrak{g}} \in [0,1]^{\hat{\varrho}}$ encodes a $\hat{\varrho}$-dimensional discrete probability distribution over the data clusters. To employ the weights in the generalized online GP modeling we need to transform the cluster weights into sample leverages. Therefore, we formally expand the vector $\mathbf{p}_{\mathfrak{g}}$ to a leveraging vector $\mathbf{b}_{\mathfrak{g}} \in \mathbb{R}^\nu$, that leverages the samples in $\mathcal{D}_\nu$ about the weight of the corresponding clusters. For a given unique partitioning of the samples in $\mathcal{D}_\nu$ into the clusters $\uplus_{j=1}^{\hat{\varrho}} \mathcal{C}_j$ the cluster index $j \in \{1, \ldots, \hat{\varrho}\}$ is known for every captured rigid motion $\mathfrak{g}_i$, $i = 1, \ldots, \nu$. Hence, we obtain the leverage parameters

$$\mathbf{b}_{\mathfrak{g}_i} = \sum_{j=1}^{\hat{\varrho}} \mathbf{p}_{\mathfrak{g}_j} \tilde{\delta}_{ij} \text{ for all } i = 1, \ldots, \nu, \tag{4.26}$$

where $\tilde{\delta}_{ij}$ defines a modified Dirac delta function which depend on the sample's cluster affiliation

$$\tilde{\delta}_{ij} = \begin{cases} 1, & \text{for } \mathfrak{g}_i \in \mathcal{C}_j \\ 0, & \text{otherwise.} \end{cases} \tag{4.27}$$

To encode the variable importance of the training samples into the generalized GP model, we present a generalized kernel function for inputs in $SE(3)$, which influences the rigid motion produced by GP prediction. The vector of sample leverages $\mathbf{b}_{\mathfrak{g}}$ is required for the new leveraged generalized squared exponential kernel function.

Theorem 4.4.1 *The function* $k_{\text{se:lev}} : SE(3) \times SE(3) \times \mathbb{R}^\nu \to \mathbb{R}_0^+$,

$$k_{\text{se:lev}}(\mathfrak{g}_i, \mathfrak{g}_j, \mathbf{b}_{\mathfrak{g}}) := \cos\left(\frac{\pi}{2}(\mathbf{b}_{\mathfrak{g}_i} - \mathbf{b}_{\mathfrak{g}_j})\right) k_{se}(\mathfrak{g}_i, \mathfrak{g}_j), \tag{4.28}$$

where we consider a valid squared exponential kernel function k_{se} *defined on* $SE(3)$, *such as* (3.44) *or* (3.46), *and* $\mathbf{b}_{\mathfrak{g}} \in \mathbb{R}^\nu$ *the vector of sample leverages, defines a valid kernel function.*

Proof 4.4.1 *We have proven already in Section 3.4.3 that the functions* $k_{se:mag}$ *and* $k_{se:wmag}$ *define valid kernel functions. As we know from [66, Chap. 6.2] that a finite product of kernels provides a kernel, it hence suffices to show that the factor* $\cos\left(\frac{\pi}{2}(\mathbf{b}_{\mathfrak{g}_i} - \mathbf{b}_{\mathfrak{g}_j})\right)$ *is positive semi-definite.*

According to Bochner's Theorem [101, Theorem 1] a stationary kernel function is a valid kernel, if its Fourier transform is non-negative; which is well-known to be non-negative for the cosine. Hence, the first factor of (4.28) *is positive semi-definite and thus, the proof is complete.* □

A more comprehensive derivation of a leveraged kernel function in Euclidean space can be found in [101]. In this paper, however, input data on the manifold $SE(3)$ is not considered.

4.4.2 Initialization

For the set $\mathcal{D}_\nu$ with ν given training data pairs the generalized online GP regression is fully described for each dimension $l = 1, \ldots, n$ by the variables mean $\mu_\nu^{(l)}$, covariance $\Sigma_\nu^{(l)}$, and the inverse of the Gram matrix $\mathbf{S}_\nu^{(l)} := \mathbf{K}_\nu^{(l)^{-1}}$, which we introduce to save the computational effort of matrix inversion.

We initialize those variables for a given first observation pair $\left\{(\,\mathfrak{g}_1,\,\dot{\mathfrak{g}}_1^{(l)})\right\}$, by setting

$$\begin{aligned}
\mu_1^{(l)} &= \frac{\dot{\mathfrak{g}}_1^{(l)} k_{\text{se:lev}}(\,\mathfrak{g}_1,\,\mathfrak{g}_1)}{\sigma_n^2 + k_{\text{se:lev}}(\,\mathfrak{g}_1,\,\mathfrak{g}_1)} \\
\Sigma_1^{(l)} &= k_{\text{se:lev}}(\,\mathfrak{g}_1,\,\mathfrak{g}_1) - \frac{k_{\text{se:lev}}(\,\mathfrak{g}_1,\,\mathfrak{g}_1)^2}{\sigma_n^2 + k_{\text{se:lev}}(\,\mathfrak{g}_1,\,\mathfrak{g}_1)} \\
\mathbf{S}_1^{(l)} &= \frac{1}{k_{\text{se:lev}}(\,\mathfrak{g}_1,\,\mathfrak{g}_1)}\,,
\end{aligned} \tag{4.29}$$

where $k_{\text{se:lev}}$ as defined in Theorem 4.4.1 and σ_n^2 denotes the variance of the noise process $\mathbf{n}_{\mathfrak{g}}$ defined below (2.10).

4.4.3 Update Step

Given the recursive variables $\mu_\nu^{(l)}$, $\Sigma_\nu^{(l)}$ and $\mathbf{S}_\nu^{(l)}$ (as initialized in (4.29)) and a new observation $\mathfrak{g}_{\nu+1}$ we seek to infer the next set of variables $\mu_{\nu+1}^{(l)}$, $\Sigma_{\nu+1}^{(l)}$ and $\mathbf{S}_{\nu+1}^{(l)}$, which fully describes the generalized online GP with $\nu+1$ observations. To enable the *prediction step*, we need to determine a scalar predictive mean value $\hat{\dot{\mathfrak{g}}}_{\nu+1}^{(l)}$ and variance $\hat{\sigma}^2_{\dot{\mathfrak{g}}(l)\nu+1}$ for each of the unknown output components $\dot{\mathfrak{g}}_{\nu+1}^{(l)}$. Then, we update the posterior Gaussian distribution in the *model update step.*

Prediction In order to keep the final equations simple and readable, four auxiliary variables are defined

$$\begin{aligned}
\mathbf{s}_{\nu+1} &:= \mathbf{S}_\nu^{(l)}\,\mathbf{k}_{\nu+1} \\
\gamma_{\nu+1}^2 &:= k_{\text{se:lev}}(\,\mathfrak{g}_{\nu+1},\,\mathfrak{g}_{\nu+1}) - \mathbf{k}_{\nu+1}^\top\,\mathbf{S}_\nu^{(l)}\,\mathbf{k}_{\nu+1} \\
\varsigma_{\nu+1} &:= \Sigma_\nu^{(l)}\,\mathbf{s}_{\nu+1} \\
\sigma_{f_{\nu+1}}^2 &:= \gamma_{\nu+1}^2 + \mathbf{s}_{\nu+1}^\top\,\varsigma_{\nu+1},
\end{aligned} \tag{4.30}$$

where $\mathbf{k}_{\nu+1}$ is a column vector with $(\mathbf{k}_{\nu+1})_i = k_{\text{se:lev}}(\,\mathfrak{g}_i,\,\mathfrak{g}_{\nu+1})$. Each of the n components of the predictive mean and variance are computed by

$$\begin{aligned}
\hat{\dot{\mathfrak{g}}}_{\nu+1}^{(l)} &= \mathbf{s}_{\nu+1}^\top \mu_\nu^{(l)} \\
\hat{\sigma}^2_{\dot{\mathfrak{g}}(l)\nu+1} &= \sigma_n^2 + \sigma_{f_{\nu+1}}^2
\end{aligned} \tag{4.31}$$

for $l = 1, \dots, n$. They are joint into a single multidimensional predictive distribution for the input $\mathfrak{g}_{\nu+1}$ via

$$\mathcal{N}\left((\,\hat{\dot{\mathfrak{g}}}_{\nu+1}^{(1)}, \dots,\,\hat{\dot{\mathfrak{g}}}_{\nu+1}^{(n)})^\top, \operatorname{diag}(\hat{\sigma}^2_{\dot{\mathfrak{g}}(1)\nu+1}, \dots, \hat{\sigma}^2_{\dot{\mathfrak{g}}(n)\nu+1})\right). \tag{4.32}$$

Model Update After observing the real output components $\dot{\mathfrak{g}}_{\nu+1}^{(l)}$ the recursive variables $\mu_\nu^{(l)}$, $\Sigma_\nu^{(l)}$ and $\mathbf{S}_\nu^{(l)}$ are updated to the posterior distribution for $(\nu+1)$-th observation via estimation error reduction

$$\begin{aligned}
\mu_{\nu+1}^{(l)} &= \begin{pmatrix} \mu_{\nu}^{(l)} \\ \hat{\dot{\mathfrak{g}}}_{\nu+1}^{(l)} \end{pmatrix} + \frac{\dot{\mathfrak{g}}_{\nu+1}^{(l)} - \hat{\dot{\mathfrak{g}}}_{\nu+1}^{(l)}}{\sigma_{f_{\nu+1}}^2} \begin{pmatrix} \varsigma_{\nu+1} \\ \sigma_{f_{\nu+1}}^2 \end{pmatrix} \\
\Sigma_{\nu+1}^{(l)} &= \begin{pmatrix} \Sigma_{\nu}^{(l)} & \varsigma_{\nu+1} \\ \varsigma_{\nu+1}^{\top} & \hat{\sigma}_{f_{\nu+1}}^2 \end{pmatrix} - \frac{1}{\hat{\sigma}_{\dot{\mathfrak{g}}(l)\nu+1}^2} \begin{pmatrix} \varsigma_{\nu+1} \\ \hat{\sigma}_{f_{\nu+1}}^2 \end{pmatrix} \begin{pmatrix} \varsigma_{\nu+1} \\ \hat{\sigma}_{f_{\nu+1}}^2 \end{pmatrix}^{\top} \qquad (4.33) \\
\mathbf{S}_{\nu+1}^{(l)} &= \begin{pmatrix} \mathbf{S}_{\nu}^{(l)} & \mathbf{0} \\ \mathbf{0}^{\top} & 0 \end{pmatrix} + \frac{1}{\gamma_{\nu+1}^2} \begin{pmatrix} \mathbf{s}_{\nu+1} \\ -1 \end{pmatrix} \begin{pmatrix} \mathbf{s}_{\nu+1} \\ -1 \end{pmatrix}^{\top} .
\end{aligned}$$

4.4.4 Characteristic Properties and Application

The generalized online GP allows for the same precision as its offline pendant. Given a set of training data $\mathcal{D}_{\nu}$, a set of hyperparameters $\mathbf{h}$ in kernel (4.28) and a new input value $\mathfrak{g}_{\nu+1}$, the generalized GP model yields the same predictive distribution as obtained by mean and variance defined in (4.31), which can be verified by simple calculations. Further, while the standard GP has complexity $\mathcal{O}(h^3)$, as it is governed by matrix inversion, the recursive GP algorithm reduces the computational complexity to $\mathcal{O}(h^2)$. As observations are added consecutively to the training data, the algorithm is suitable for learning the GP model online. For application in practice, however, the recursive GP requires the following adaptations:

Fixed-Budget Online GP

We denote by dictionary the set of training data on which the next prediction is made. So far, the recursive GP has an ever-growing dictionary. In order to limit the requirements of computation and memory we introduce the fixed-budget online GP, possessing a limited dictionary size. In case a predefined budget β is exhausted, i.e. on obtaining $(\beta+1)$-th observation, we select and remove the least relevant pair $(\mathfrak{g}_j,\, \dot{\mathfrak{g}}_j) \in \mathcal{D}_{\beta+1}$. The index j is determined by a cost function measuring the information loss caused by removal of the pair

$$j = \arg\min_{i} \left(\frac{\left(\mathbf{S}_{\beta+1}^{(l)} \mu_{\beta+1}^{(l)} \right)_i}{\left(\mathbf{S}^{(l)} \right)_{i,i}} \right)^2 \qquad (4.34)$$

Then, the operators $(\cdot)_{-i}$ and $(\cdot)_{-i,-i}$, which remove the i^{th} row in a vector and both the i^{th} row and column in a matrix, respectively, are applied to remove the corresponding information from the regression variables. Hence, by removal of j-th observation

$$\begin{aligned}
\mu_{\beta+1}^{(l)} &= \left(\mu_{\beta+1}^{(l)} \right)_{-j} \\
\Sigma_{\beta+1}^{(l)} &= \left(\Sigma_{\beta+1}^{(l)} \right)_{-j,-j} \qquad (4.35) \\
\mathbf{S}_{\beta+1}^{(l)} &= \left(\mathbf{S}_{\beta+1}^{(l)} \right)_{-j,-j} - \frac{\left(\mathbf{S}_{\beta+1}^{(l)} \right)_{-j,j} \left(\mathbf{S}_{\beta+1}^{(l)} \right)_{-j,j}^{\top}}{\left(\mathbf{S}_{\beta+1}^{(l)} \right)_{j,j}}
\end{aligned}$$

we re-obtain in the $(\beta+1)$-th recursion step a dictionary of size β.

Forgetting Strategy

Under non-stationary working circumstances the generalized recursive GP additionally needs a strategy to "forget" past samples to be able to adapt to a varying underlying mapping [100]. The basic idea of the forgetting strategy is to slightly reset the main vector and the covariance matrix to their initial values after each kernel updates. A forgetting factor $\gamma \in (0, 1]$ is applied to the mean and covariance of the generalized online GP by

$$\begin{aligned} \Sigma_\nu^{(l)} &= \gamma \Sigma_\nu^{(l)} + (1-\gamma)\mathbf{k}_\nu^{(l)} \\ \mu_\nu^{(l)} &= \sqrt{\gamma}\mu_\nu^{(l)}. \end{aligned} \tag{4.36}$$

The smaller the forgetting factor γ, the faster the algorithm is able to track changes, meanwhile it learns less. When choosing $\gamma = 0$, the regression is with complete forgetting, whereas $\gamma = 1$ corresponds to no forgetting.

4.5 Experimental Evaluation

In this section we validate our approach to approximate the decision process using a toy example, namely elderly assistance in putting on shoes. Since it affects the comfort of the human shoe receiver, we desire the robotic assistance to behave human-like in its approach to the human, the shoe positioning relative to the human foot and the slipping of the shoe over the foot. Hence, we capture and analyze human-human behavior in this cooperation task to design attraction forces for a robotic control law to imitate human shoe providing behavior.

First, we capture and cluster unconstrained human leg movement, when the participant expects to be provided a shoe. Using the results of this analysis, we determine the number of shoe reception conditions $\hat{\varrho}$ in which the spatial movement plans are split in and the corresponding cluster parameters. Subsequently, we perform a human-human cooperation analysis: One participant takes over the role of the provider, the other is the receiver (representing the elderly that needs assistance). Meanwhile the receiver indicates the desired dressing pose by moving the foot towards it, the provider approaches the receiver, adapts the shoe positioning to the receivers goal and slips the shoe over the presented foot. Figure 3.7 and Figure 4.1 illustrate the procedure for different human-human pairs. Based on the provider behavior in the preliminary human-human cooperation task, we approximate the weighting algorithm for the receivers potential movement plans and obtain the leverage parameters for the data samples. The sample leverage is used for predicting the human internal spatial movement goals, which in turn are utilized as weighted attractor forces in the robot control.

A robot with mobile platform was employed to execute the shoe putting on to a sitting human participant. To allow for precise movement tracking of the human, markers were attached to the human body and the robot was equipped with a rigid grasp unit for the shoe. Figure 4.2 shows a successfully shoe dressing of a human participant by the robot. Additionally, the picture visualizes the robotic setup including the rigid grasp unit for the shoe.

The experiment (consisting of a human user study about the comfort and efficiency of the lower limb assistance by the robot) could not be finalized due to a fire accident. Therefore, the leveraged generalized online GP will not be evaluated experimentally. Instead, we

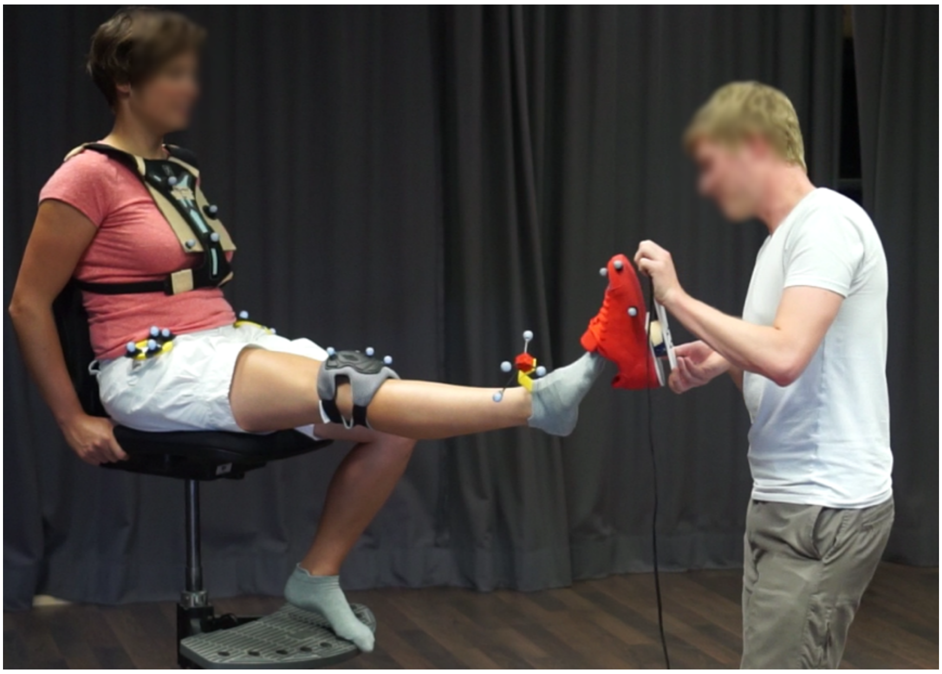


Figure 4.1: One human puts on a shoe to another in a human-human cooperation study to prepare intuitive elderly assistance in service robotics. A motion tracking system captures the provider's adaptation of the shoe to the receiver's leg motion and a 6-DoF force and torque sensor captures the contact forces during the slipping over motion.

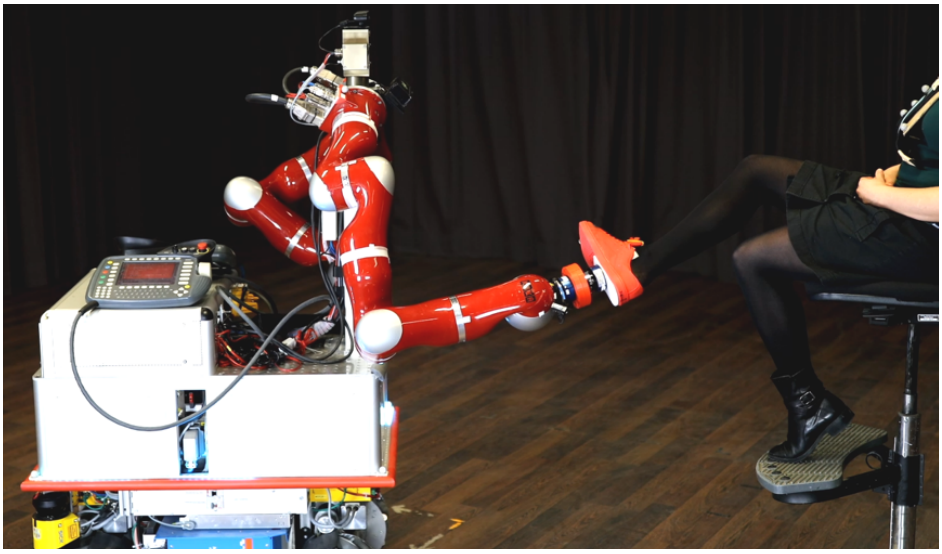


Figure 4.2: Successful robotic lower limb assistance. A robot approaches the human shoe receiver, positions the shoe relative to the estimated receiver's goal foot pose and successfully slips the red shoe over the foot.

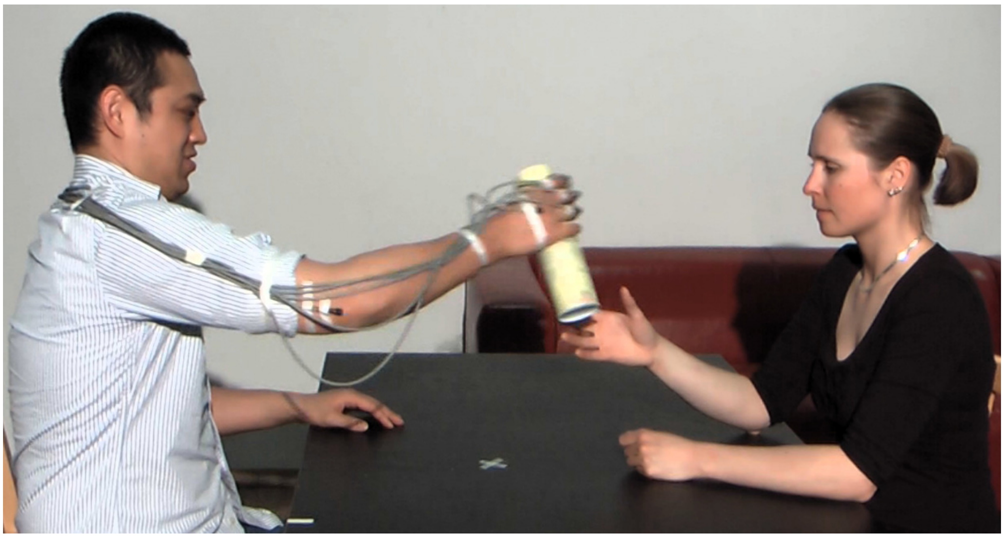


Figure 4.3: A natural 6D human handover is visualized. The object handover starts from an initial hand configuration, performs the object grasp and terminates at the final configuration after the object is handed over. The handover motion in this trial incorporates a half turn of the object.

demonstrate the generalized online GP with leverage parameters fixed to the unit vector $\mathbf{1}$ in a human-human handover scenario.

4.5.1 Generalized Online GP Demonstration

The aim of this real data experiment is to generate a human motion prediction online. Therefore, we employ the generalized online GP with a fixed leverage parameter $\mathbf{p}_{\mathfrak{g}} = 1$. We do not rely on task specific prior knowledge in our approach to guarantee maximal adequacy for arbitrary tasks, but use a totally generic initial choice for the model parameters. The example we choose in this experiment, visualized in Figure 4.3, concerns a human object handover including significant rotational motion, namely turning the object around by a rotation about π. We approximate this handover task by employing the decision process dynamics (2.10). With the generalized online GP multiple potential movement plans in $SE(3)$ can be generated in parallel. We, however, exemplary only predict one such plan and therefore, do not require alternative leverage parameters.

Experimental Conditions

A pair of human participants performed an object handover where the passers hand motion was tracked using a magnetic motion tracking system (Polhemus Liberty) with a sampling rate of 80 Hz. The subject handed over a cylinder (21 cm in length, 5.5 cm in diameter and weight 280 g). We approximate the human motion using the generalized online GP model and evaluate the results using the captured 6-DoF data as ground truth. The forgetting factor is fixed to 1, i.e. no forgetting, and the dictionary size is limited to 100 observations, as a trade-off between regression performance and computational demand. The regression is performed online on a commercially available computer with Intel core i5 processor and 8 GB RAM. One iteration of the generalized GP model incorporats the calculation of a predictive mean and variance in all 6 output dimensions plus the update of the 6-DoF posterior distribution. On average, a recursion step takes 0.38 sec.

Application of Generalized Online GP and Kernel Reset

From the captured data, we obtain the input set of dual quaternions $\{\mathfrak{g}_i\}_{i=1}^{\nu}$ representing the hand poses throughout the handover and the corresponding output velocity set $\{\dot{\mathfrak{g}}_i\}_{i=1}^{\nu}$. Each 6D velocity consists of three translation velocities $\dot{\mathfrak{g}}_x$, $\dot{\mathfrak{g}}_y$, $\dot{\mathfrak{g}}_z$ and three rotational velocities represented in the tangent bundle $\dot{\mathfrak{g}}_{TS_x}$, $\dot{\mathfrak{g}}_{TS_y}$, $\dot{\mathfrak{g}}_{TS_z}$. We approximate the dynamics (2.10) using the $\mathrm{GP}_{\mathbb{H}_D}$ over dual quaternions with squared exponential kernel $k_{\text{se:mag}}$ as introduced in (3.44) and initialize the hyperparameters with $\mathbf{h}_0 = (1, 1, 1)$. Per newly observed sample pair $(\mathfrak{g}_{\nu+1}, \dot{\mathfrak{g}}_{\nu+1})$ the training data set $\mathcal{D}_\nu$ increases until the budget limit 100 is reached and the least relevant pair removal process is additionally activated.

As it is our aim to demonstrate the wide applicability of the proposed approach besides the experimental validation, we do not perform prior investigations to find optimal hyperparameters, despite the fact that this is the common routine in GP modeling. Instead, we train the hyperparameters on the four first observed pairs $\{(\mathfrak{g}_i, \dot{\mathfrak{g}}_i)\}_{i=1}^{4}$ and perform the recursive GP over dual quaternions with this possibly suboptimal hyperparameter set. As these first poses might not be representative for the whole following motion, we accept less accurate prediction results in favor of robustness of the algorithm and a wide application range. In case the mean prediction deviates from the subsequently obtained ground truth value more than a tolerated prediction error (in this experiment we use 0.6 as threshold), we train new hyperparameters on the following four dual quaternions.

Visualization

The estimation results are illustrated in Figure 4.4. For clarity the 6D output is shown per dimension over the runtime corresponding to the sample index $i = 1, \ldots, \nu$ instead of the GP input data $\mathfrak{g}$. The vertical bar indicates the current time stamp, i.e. left of it is the past, where the Gaussian posterior is available and on its right is the future, where we estimate a predictive Gaussian. The ground truth is depicted by the red dashed line and the mean prediction by the blue solid line. The grey shaded area visualizes the 2σ confidence interval for the estimation and captures thereby the noise involved in real data. In the first and second row we depict the three dimensions of the translational velocity and tangent bundle velocity, respectively. In the third row, the rotational velocity estimation is projected to the sphere S_3 using (3.53). We visualize the three imaginary components of the resulting quaternions only. The velocity predictions, row one and two, demonstrate that in the past the mean prediction deviates little from the ground truth. The distance, how soon the mean prediction returns to zero in the future depends on the length-scale hyperparameter λ learned from four consecutive observed poses only. We indicate the kernel reset with black dots on the ground truth trajectory. In the third row, a slight asymmetry in the uncertainty estimate can be observed. It results from the projection to a curved space. The results show further that over time the prediction drifts from the ground truth as the velocity prediction error accumulates. Recall that we could significantly improve the estimation results through elaborate hyperparameter tuning, but as this contradicts the gist of online modeling, we refrain from any preliminary task specific tuning.

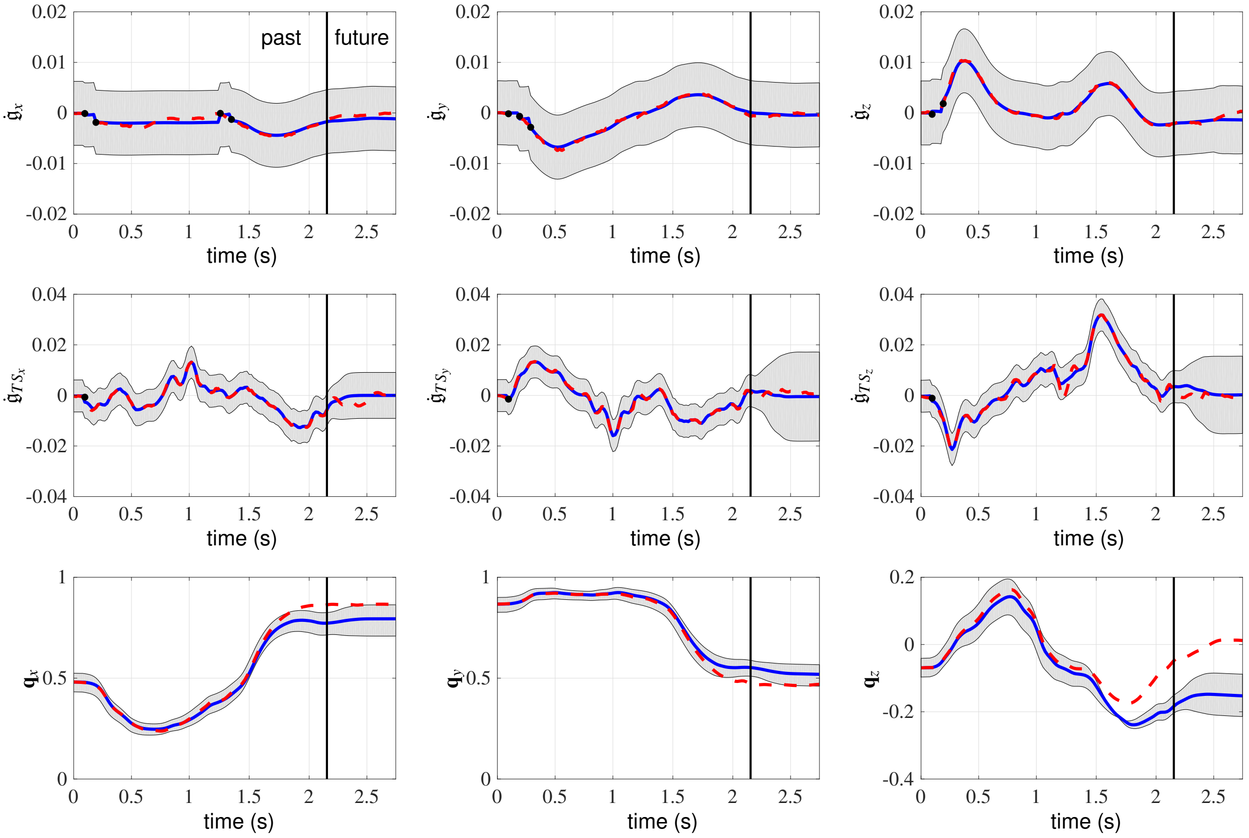


Figure 4.4: Estimation results of the generalized online GP algorithm. The vertical bar indicates the current time stamp. In the past, left of the bar, the Gaussian posterior is available, whereas on its right, in the future, the motion is predicted. The first row depicts the translational velocities, the second row the tangent space velocities and the third row shows the imaginary components of a quaternion, which is obtained by projecting the velocities of the row above to the sphere.

4.5.2 Rigid Motion Clustering of Shoe Receiver

To cluster the executed receiver's foot motions into a reasonable set of trajectory bundles, we first capture freely performed human foot presentations (for future shoe receiving) using a motion tracking system (Qualisys Sweden) with a sampling rate of 179 Hz. Five subjects (1 female, 4 male) with the average age of 31±4 years and body height 1.77±0.18 (m) performed each 100 repetitions of intuitive foot motions. The participants were only instructed to remain seated and to move their foot in a pose how they desired to be put on a shoe. Otherwise, the subjects were free to perform the foot demonstrations as they wished.

Since we aim to cluster the movements according to their spatial characteristics instead of the body characteristics of the participants, we need to scale the subjects to the size of a virtual human. Therefore, we calculate the norm of the foot position in the coordinate frame centered in the chest throughout the motion and use its maximum as scaling factor,

$$c_{\text{scale}} = \max_{\kappa \geq 0} \| \, \mathbf{v}_{\text{foot},\kappa} - \, \mathbf{v}_{\text{chest},\kappa} \|. \tag{4.37}$$

We consider 1.5 m as maximum distance of the standard virtual human and calculate the foot pose in chest frame scaled by $1.5/c_{\text{scale}}$, i.e. the translation is scaled, while the rotation remains unchanged.

We apply the generalized GMM clustering to the goal poses of the foot in chest frame. In this experiment we set the distance bound of the inequality (4.3) to $c_{\max} = 10^{300}$ and the convergence bound for the loglikelihood (4.15) to $10^{-6} \ln p(\{\, \mathfrak{g}_j \}_{j=1}^{\nu} \big| \boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\chi})$. We calculate the BIC value iteratively for increasing cluster cardinality from 1 to 10. Then, we repeat the experiment 70 times, to assure reproducible results. In our experiment we found that according to both, the lowest achieved BIC values and the average BIC values, the optimal number of clusters $\hat{\varrho} = 5$. Figure 4.5 visualizes on the left the average and standard deviation of the BIC values and on the right the optimal BIC value over the 70 experiment repetitions for each of the 1 to 10 clusters. Globally, the optimal achieved BIC value is 3286.56. Additionally, Figure 4.6 shows a histogram of the optimal achieved BIC values over all experiment runs. In 25 out of the 70 repetitions the GMM with 5 clusters turned out to be BIC optimal. The lowest obtained BIC values during the remaining 45 repetitions spreads over the cluster cardinalities $3 - 8$.

To proof that the optimal number of clusters (in the present case 5) only coincidentally equals the number of subjects, we additionally visualize the distribution of clusters among the subjects in Figure 4.7. The female subject has number 3, the shortest male subject has number 5 and the tallest male subject has number 2. Hence, no correlation between the subject's body size or the gender and the clusters is discernible.

The clusters of the generalized GMM model with the globally lowest BIC value, i.e. BIC = 3286.56, are visualized in Figure 4.8. The human shape represents the standardized virtual human with maximal foot to chest distance of 1.5 m. Since, we represent the foot motion in chest frame of the human, as depicted in the figure, the goal positions of the foot, visualized by colored circles in the space, depend on the posture of the participant. Hence, especially the chest frame orientation is affected by whether the subject sits upright or crunched on the chair. Therefore, some of the goal positions might seem out of reach for the exemplary visualized human shape. The color code of the samples shows the cluster assignment for the samples selecting for each the cluster with highest generalized GMM probability. We only visualize the positions of the samples, to not superpose the data clusters with 500 cluttered

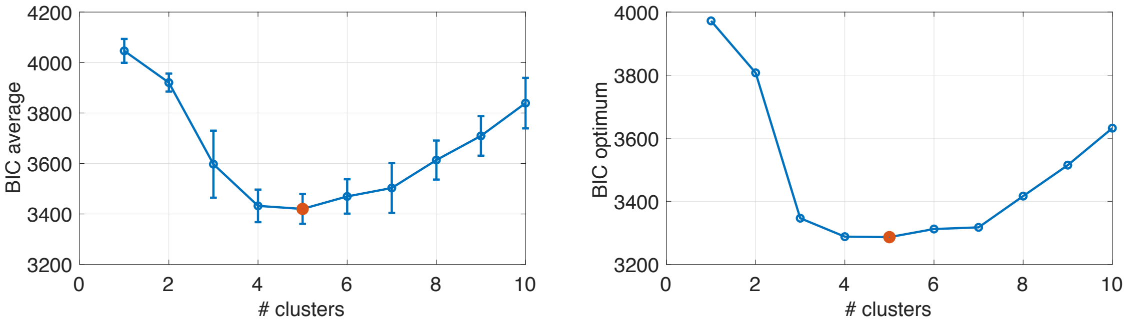


Figure 4.5: Bayesian information criterion (BIC) value for GMM with cluster cardinality form 1 to 10. On the left, the average and standard deviation of the BIC values over 70 experiment repetitions are depicted; on the right the lowest BIC values obtained throughout all 70 experiments are shown for each cluster size. In both settings, the optimal number of GMM clusters $\hat{\varrho}$, depicted in red, is 5.

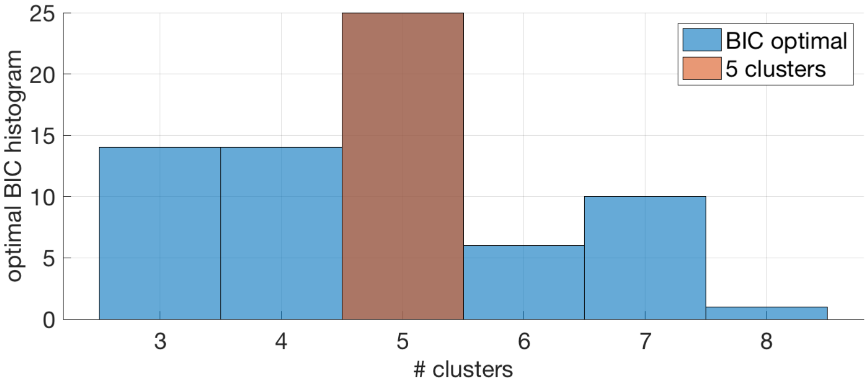


Figure 4.6: Histogram of the optimal obtained BIC values over 70 experiment repetitions. Clearly, the GMMs with 5 clusters generated the lowest BIC value in most cases (25 out of 70).

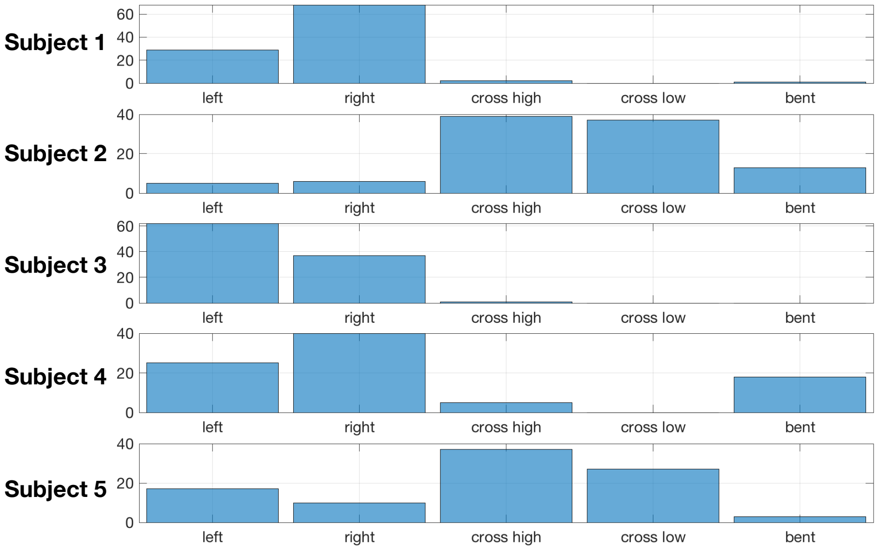


Figure 4.7: Histogram over the final foot clusters performed by each of the 5 subjects. The distribution over the goal configurations shows the diversity in personal preference of the individuals but no correlation between the clusters and personal attributes of the subjects.

coordinate frames. For each generalized GMM component, however, one coordinate frame in the center of the cluster shows the main axes of the fitted projected Gaussian distribution. The generalized GMM covers well the reachable space of foot goal poses of the shoe receiver and contains most of the data set within the Gaussian 2σ-uncertainty regions, depicted by the transparent cluster ellipsoids.

4.5.3 Classification Analysis in Comparison with Humans

As we aim to design in future work a human-behavior-based control for the robot to assure intuitive assistance, we require a model for the human motion behavior in the specific task of passing and slipping on a shoe to another one's foot. Hence, we carry out a human-human interaction study to estimate the required parameters to imitate human behavior from the preliminary interaction study. In seven teams of one shoe provider and one shoe receiver, we analyze intuitive human interaction behavior. Based on the findings of the study, we set up the decision making among the potential movement plans using dynamics (2.10) and compare our algorithm to human provider ability to identify the aspired receiver's class. The presented decision process algorithm outperforms the human in speed and accuracy.

Experiment Setup

Every experiment repetition is conducted as follows. The shoe receiver sits on a chair with both feet flat on the ground. The shoe provider stands approximately 2.5 m apart and turned away, so that he/she cannot see the receiver. After a signal, the provider picks up the shoe, which was put on a table in front of the provider, and turns towards the receiver. Only after the receiver had visual contact with the provider, he/she initiates the own foot movement. The receiver moves the foot towards one of the goal configurations, that were previously identified in the clustering experiment in Section 4.5.2. The provider approaches

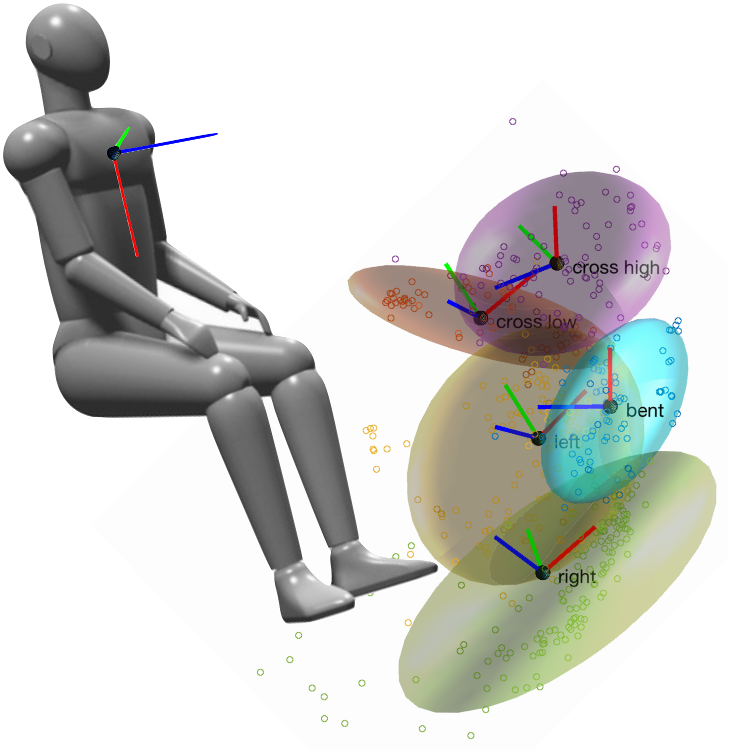


Figure 4.8: Generalized GMM clusters for which we obtained the globally optimal BIC value 3286.56. The receiver's foot goal position is illustrated by circles, colored according to the most probable cluster membership. They are presented in human chest frame, and thus, depend on the receiver's sitting posture. Even though only the data positions are visualized for clarity reasons, the generalized GMM clustering was performed on the full rigid motions $\mathfrak{g}_j$, $j = 1, \ldots, \hat{\varrho}$. The ellipsoids represent the fitted cluster components. Each projected Gaussian is centered at one of the coordinate frames *right, left, bent, cross low, cross high* and spans the corresponding ellipsoid with it's 2σ-uncertainty region.

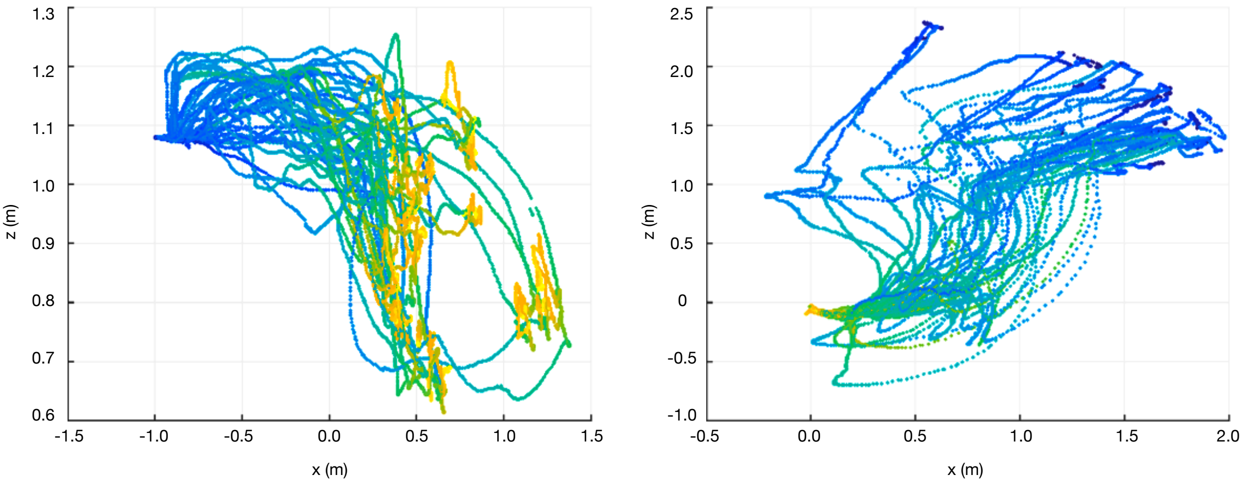


Figure 4.9: Trajectories of the shoe in 3D space. The color gradient of the trajectories depict the evolution of the movement from the start (dark blue) to the goal configurations, when the foot is successfully inserted in the shoe (yellow). The same set of trajectories, which is represented in a global frame on the left, is represented in the goal foot frame on the right. The goal foot frame is the configuration, in which the shoe insertion in that specific cluster has taken place.

the receiver and slips the shoe over the foot, which in the meanwhile arrived in the goal configuration.

The goal configurations are only known to the receiver, not to the provider. Every pair of receiver and provider repeats the experiment 65 times with an priori set random permutation of a sequence of uniformly distributed goal configurations, i.e. each of the 5 shoe putting on configurations is conducted 13 times. The first 5 repetitions are considered as training for the provider to get to know the possible classes. In the subsequent 12 repetitions per condition, randomly 2 executions are delayed by the shoe receiver for a couple of seconds to analyze the waiting behavior of the provider.

Human Behavior Analysis

In this section we analyze the behavior of the shoe provider to approximate the decision process analogously to the human decision making, as human-like robotic behavior allows for intuitive human-robot cooperation. We investigate the characteristics of the provider's shoe movement in terms of trajectories. Our findings show that the probabilistic trajectory classification, presented in Section 4.3, exhibits the required characteristics to estimate the weighting of potential spatial plans and the induced decision making. The details are provided in the following.

Figure 4.9 demonstrates a set of shoe trajectories in 3D space, performed by various shoe providers. The rotation is omitted for clarity of the visualization. All trajectories start from a similar initial configuration, as the shoe is placed on a table in the beginning, and are cut on the onset of the shoe movement. The trajectories end when the shoe is successfully put onto the receiver's foot for any of the 5 optional clusters of final configurations. The trajectories are color coded: From the motion onset (in dark blue) the color gradient is getting brighter to the trajectory's offset with the shoe on the receiver's foot (marked in

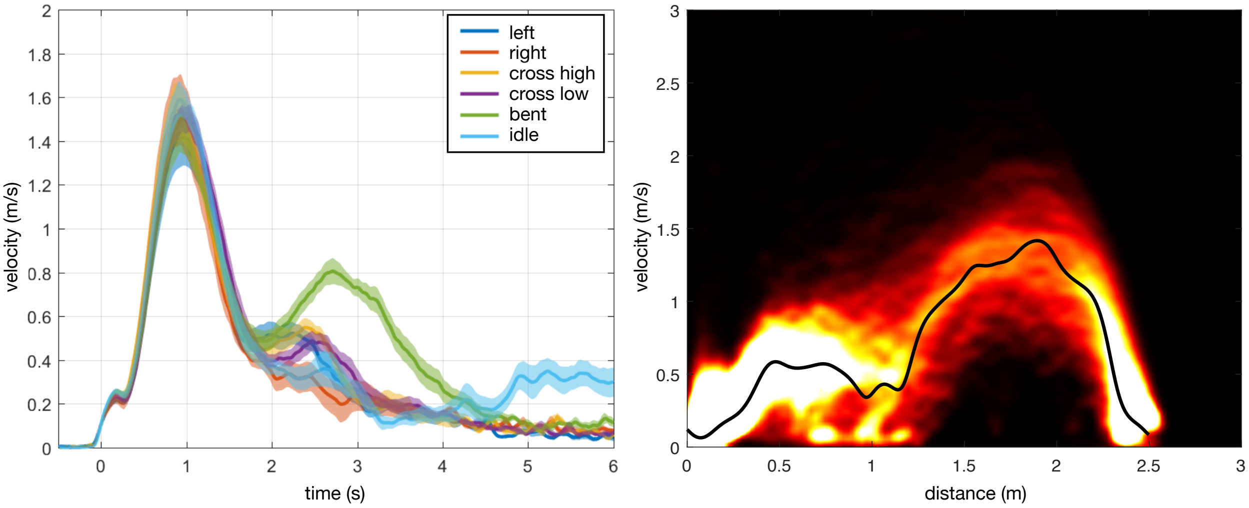


Figure 4.10: Norm of the translational velocity trajectories of the shoe. On the left side, the mean and the standard deviation of the velocities per condition are visualized over time, and on the right side, the variability in the execution velocity is depicted over the distance to the foot.

yellow on the trajectory). In the left plot of Figure 4.9 the trajectories are shown in a global frame and the various final configurations can be recognized as 5 clusters of yellow regions in translational space. Note, the cluster membership gets more obvious when one takes rotation into account, which in this visualization is absent for the sake of simplicity. In the right plot of Figure 4.9 the trajectories are represented in the final foot frame. Hence, the goal configurations of each cluster are superimposed, and thus, the yellow ends of the trajectories are joint in a similar final configuration in the goal foot frame.

The overlay of shoe trajectories demonstrates a similar provider movement at the beginning and at the end of each experiment repetition (even though the trajectories are captured from different providers). We interpret this behavior like this: In the initial phase where the decision making has not yet terminated, no adaptation to the specific shoe insertion configuration is present. Likewise, the movements show a standardized approach within a foot neighborhood of $0.1 - 0.2\ m$, and the insertion of the foot into the shoe follows the exact same pattern for all movement clusters. Hence, the decision making and the adaptation of the shoe to the foot goal configuration has terminated before this final phase.

The assumption that the provider's behavior initially is a standardized motion towards the receiver is strengthened by the clear pattern in the velocities for all experiment repetitions. Figure 4.10 shows the translational velocity norms over all captured shoe movements (all conditions and participant pairs). The norm is employed to unify the 3D translational velocities. In the left plot the mean and the standard deviation of the velocities per condition are visualized over time, while in the right plot the variability in the execution velocity is depicted over the distance to the foot. The velocities were calculated using zero-order hold (i.e. the difference quotient) from the shoe movement trajectories, synchronized on the onset.

The pattern in the velocities on the left of Figure 4.10 can be interpreted as follows. The provider continuously moves the shoe towards the receiver. Even though the motion is slowed down during decision making, the movement towards the expected goal is continued. When a precise goal configuration of the foot can be anticipated, the provider positions the shoe relative to the estimated foot configuration while adapting online to the receiver's motion. The small captured velocities towards the end of the trajectories, originate from the drift in the human motor system and from measurement noise. Note that taking the norm of the velocities additionally amplifies the appearance of the noise. We would like to point out that the second peak in the velocities of the *bent* condition (drawn in green) are due to the particular large distance of the provider and the foot in this condition. Besides, in the idle condition, where the receiver did not move the foot, the provider comes to rest (roughly in the time interval $3 - 4\ s$) while waiting for the receiver to move his/her foot.

On the right of Figure 4.10 the shoe trajectory has to be understood from right to left. When the movement is initiated, the distance between the shoe and the foot is largest (about $2.5\ m$); when the experiment repetition terminates, the distance between foot and shoe is circa $0\ m$, since the foot is inserted in the shoe. The visualization clearly shows for distances larger than $1.2\ m$ a phase of standardized approach of the provider towards the receiver. The phase when no single condition is clearly prioritized is characterized by low motion velocities (around $1\ m$ distance to the shoe). And finally, the third phase of the shoe providing executes the demanded adaptation of the shoe to the foot.

Hence, we deduce from the human-human study, that a fast and reliable probabilistic classification algorithm is required to approximate the potential movement weighting in the decision process.

Evaluation of the Classification Algorithm

In this section we evaluate the speed and accuracy of the probabilistic trajectory classification algorithm as introduced in Section 4.3. The algorithm is trained per condition on 15 trajectories $\{\,\mathfrak{g}(t)\}_{t\geq 0} \subset SE(3)$ (consisting of positions and orientations), i.e. in total on 75 motion trajectories of the receiver's foot. Then, the algorithm calculates online the class-probabilistic likelihood (4.22) from unseen human receiver trajectories and estimates the goal cluster using the classification rules introduced in Section 4.3.2. We test the algorithm on 10 unseen trajectories per condition, which start with the motion onset. The algorithm classified 42 of the 50 test trajectories correctly after 47.08 ($\pm$6.48)% of the trajectory length on average ($\pm$ standard deviation). Table 4.1 details the classification results per condition. In most cases the classification terminated within the first second after the motion onset, which is fast. The false classifications seem to slow down the classification time in some cases (see for instance the conditions *cross low* and *right*). However, occasionally the false classifications appear "rushed" (e.g. the wrong decision in the *bent* condition is made significantly earlier, all false classifications for condition *left* are faster than the correct ones and in the *right* class the large standard deviation suggests partially early classifications).

The testing trajectories are cut to the rigid motion $\mathfrak{g}(T)$ when the classification has terminated. We synchronize the test movement trajectories with video data from the receiver's foot. Accordingly, the videos are cut to the last video frame before the onset of the motion and the first video frame after the algorithm finished the classification. Thus, the video snippets contain two frames more than the information the algorithm had for the decision making.

Table 4.1: Overview over the classification results broken down according to the conditions.

Conditions / Classification	cross low	cross high	left	right	bent
success rate (%)	80	100	80	70	90
duration mean(std) in s	0.95(0.14)	0.97 (0.18)	0.85 (0.07)	1.04 (0.21)	0.92 (0.12)
duration mean(std) in % of traj. length	47.00(6.62)	48.26(9.56)	41.95(3.76)	52.37(11.18)	46.00(6.05)
duration mean(std) in % for success	46.45(4.95)	48.26(9.56)	42.37(3.39)	51.28(4.77)	47.25(4.86)
duration mean(std) in % for failure	49.21(14.51)	- (-)	40.26(6.33)	54.91(21.91)	34.74(0)

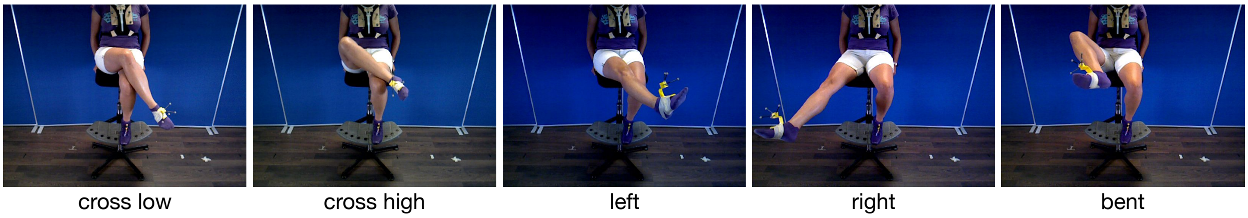


Figure 4.11: Goal configurations of the 5 clusters *left, right, cross high, cross low* and *bent.* The pictures are shown to the participants as representatives of the possible clusters.

Evaluation of the Human Decision Making

In this section we evaluate the speed and accuracy of humans in movement classification. The movement patterns of each of the 5 different clusters are demonstrated to the human participants. Then, the participants are asked to identify the corresponding cluster *left, right, cross high, cross low* or *bent* of a random set of 100 test video sequences. The set of 100 test movements is sampled uniformly over the 5 conditions. In this experiment, however, we exclude the test trajectories where the algorithm misclassified, to avoid impeded circumstances for the humans due to rushed decision making of the algorithm (and thus, extremely short video snippets). After having watched the whole video sequence, the participant is asked to select the class in which they believe the presented movement will terminate. Therefore he/she chooses a goal configurations presented in Figure 4.11. The experiment is designed this way to not pressure the participant to speed up, since time pressure might result in careless mistakes. Additionally, we do not have to deal with delays of the participants due to reaction time.

The experimental is evaluated on 11 subjects (6 male, 5 female). The average age (± standard deviation) is 36.1 (±16.6) years. The youngest subject is 5 years old, the oldest 65. In 57.36% the participants predict the correct goal configuration with a standard deviation across the subjects of ±12.52%. Differences among male and female participants are not significant; accuracy of males: 57.17% (±11.60%) versus accuracy of females:

55.40% (±17.95%). Age dependency as well can not be approved. Even the 5 year old child performed middling. Table 4.2 details the classification results per condition. Even though the conditions are sampled uniformly at random, the law of large numbers allows quite large deviations among the condition frequency. The condition *bent* is sampled 258 times, whereas the condition *right* only 190 times in total for all participants. The subjects tend to predict the *cross* conditions considerably more often than the other conditions. This, however, does not mean, they perform particularly well on those conditions. In fact, the participants perform best on the *right* condition. In 15.36% out of the 18.45% when the condition was sampled, the humans correctly identify this condition. Recognizing the *left* condition is apparently most difficult. The subjects correctly predicted it in only 5.18%. However, this might be due to the participants predicting this condition scarcest.

Table 4.2: Overview over all 1100 repetitions broken down according to the conditions.

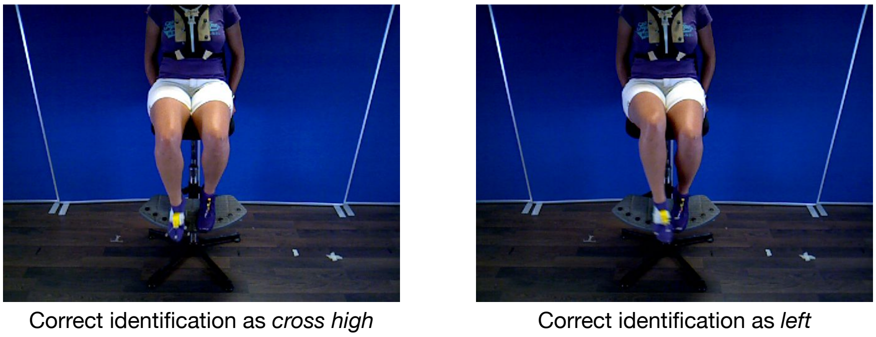
Conditions / Evaluation (in %)	cross low	cross high	left	right	bent
How often randomly sampled	20.09	20.64	18.55	17.27	23.45
How often predicted	28.27	24.45	13.27	18.45	15.54
How often correctly predicted	13.18	11.00	5.18	15.36	12.64
How often wrongly predicted	15.09	13.45	8.09	3.09	2.90
How often wrongly not predicted	6.91	9.64	13.36	1.91	10.81

Contrasting the Algorithm against Human Decision Making

The algorithm clearly outperforms human classification abilities. The best performance that is achieved by the humans is 80% correct classification, while the algorithm achieves 84%. On average, the human participants only achieve 57.36% correct classification. The participants unanimously complain about not having come to a decision about the goal configuration within the short video sequence. Hence, we demonstrate that the algorithm is faster and more accurate than humans in the decision making, even though we provide more information to the human than to the algorithm.

Figure 4.12 shows two different human movement trajectories that are correctly classified into the conditions *cross high* (on the left) respectively *left* (on the right). The pictures are the last video frames of the video snippets; when the algorithm had (correctly) classified each of the two motions into the clusters. For humans it is hardly possible to determine the correct condition in such an early stage of the motion, even though in the present pictures the differences in the whole leg posture are clearly visible in the direct comparison.

Remark 4.5.1 *I participated in this experiment and were not able to outperform the algorithm. Even though I had developed and implemented the algorithm and had executed the recorded movements, I only achieved the human record of* 80% *correct classification. If I would be excluded from the experiment, the performance of the remaining 10 participants would drop to* 55.10% *correct predictions on average with a standard deviation of* ±10.56%.



Correct identification as *cross high*

Correct identification as *left*

Figure 4.12: The first video frames after the online classification had correctly classified the synchronized movement trajectories into the conditions *cross high* respectively *left*. The algorithm is based on the foot motion only and does not take the knee movement into account.

4.6 Discussion

To conclude this chapter, we review our achievements concerning the approximation of the human decision process, which consists of a parallel process of potential movement planning and weighting of the potential plans.

As we do not have access to the human internal potential desired trajectories, we rely on executed human behavior in similar situations, where the subjects had to achieve the same task goal. We provide an algorithm for probabilistically differentiating the human movement plans, namely a generalization of the GMM clustering to rigid motion input data. Using this approach to estimate the human internal movement plans, we consider variable data densities and heterogeneous cluster sizes, as well as the uncertainty in human motor execution by modeling the clusters with Gaussians. Additionally, the EM-algorithm, which is employed to fit the GMM to the data, autonomously handles the required number of clusters. In an experimental evaluation of the algorithm, we validate the GMM clustering for human shoe putting on for different configurations on $SE(3)$. According to the BIC measure, we receive $\hat{\varrho} = 5$ as optimal estimate of the number of goal configurations. Therefore, we bundle the corresponding trajectories accordingly to approximate the human internal desired trajectories $\{\, \mathfrak{g}_{\mathrm{d}}^{1}(t)\}_{t\geq 0}, \ldots, \{\, \mathfrak{g}_{\mathrm{d}}^{5}(t)\}_{t\geq 0}$ in this specific task.

Further, we introduce a probabilistic classification for rigid motion trajectories to weight the movement alternatives. This algorithm is based on a point wise similarity measure between a new unseen motion trajectory and the approximated potential movement plans. We develop a suitable similarity measure based on the concept of the Mahalanobis distance, but allowing for rigid motion input. Due to the point wise calculations, the weighting operates online on the new unseen trajectory, until the algorithm has come to a decision for the classification through applying the presented classification rules. We evaluate the online classification algorithm in comparison to human decision making abilities in the task of predicting the goal configuration of human foot movement. Our investigations show, that the algorithm accomplishes this task faster and with higher accuracy than humans. Since the algorithm is online capable, it generates a probability distribution encoding the probabilities of the potential spatial movement plans to be assigned as underlying desired trajectory for each captured measurement.

The simultaneously running process for predicting the ongoing new movement trajectory incorporates the potential movement weighting. We introduce in this chapter the procedure, how the cluster weights are used to leverage the training samples for the online trajectory prediction. The GP kernel function is modified to account for the sample leverages. Further, the GP is implemented recursively to allow for online model training and prediction on rigid motions. Hence, we introduce a generalized online GP including sample leveraging. We validate the generalized online GP on real human motion data. However, in the presented experiment, the sample leverages are fixed to 1.

We are looking forward to employing our decision process approximation as a whole to the task of putting on a shoe to some human by a robot, and further applications in elderly care and assistance. This application field is growing due to the demographic change in many countries world wide. The idea of parallel behavior planning and weighing the options, however, is not restricted to certain application domains, but can be applied to a wide variety of human modeling tasks. It does not even necessarily involve movement data. It could for instance analogously be applied to natural language domains for conversation topic estimation.

5

Ambient Symptom Recognition in Parkinson's Disease using a Wearable Tracking Device

In this chapter we present our approach of estimating the human internal motor state dynamics (2.11) along the example of human motor symptom estimation of Parkinson's disease (PD) patients. In Section 2.3 we introduced an external monitoring system to estimate the motor state, since we have no access to the human internal motor states. Therefore, we employ the underlying assumption of the standard internal motor state rating procedure in PD [62]: We identify the patient's internal motor state based on the intensity of motor abnormalities. A fault detection system (2.16) can then detect the share of motor dysfunction in executed human motor behavior and estimates the motor symptoms. Recall, that we denote the motor symptoms and the side effects of medication, both with "symptoms".

In the reminder of this chapter we detail our technique for the motor symptom estimation. Therefore, we introduce our method for monitoring the motor symptoms of Parkinson's patients by stochastically modeling the relationships between their wrist movements during unscripted daily activities and corresponding annotations about clinical displays of movement abnormalities. We approach the estimation of PD motor behavior by multiple GPs for three classes of commonly observed movement abnormalities in patients with PD including bradykinesia with tremor, bradykinesia without tremor, and dyskinesia. A special characteristic of our method is that it not only estimates the presence of those motor symptoms, but also their severities.

This chapter is structured as follows. We start with presenting important background information and facts about PD. Then, we introduce the monitoring system, i.e. our methods for data acquisition. Subsequently, the methodology for approximating the human motor symptom is detailed, consisting of feature identification in the monitoring unit and motor abnormality recognition and estimation in the fault detection. Finally, we present our application setup for a real patient data evaluation and discuss the experimental results.

5.1 Background Information on Parkinson's Disease

PD is the second most common neurodegenerative disease after Alzheimer's disease, primarily associated with various forms of movement-related deficits [123]. As many as 60000 new cases are diagnosed every year in North America [124], and according to a recent analysis from [125] the prevalence of PD is estimated to be 217.22/100000 in Germany.

In the following, we describe common PD symptoms and how they are assessed using a standard PD rating scale. Then, relevant related work on this topic of automated PD symptom recognition is reviewed, and we highlight the remaining unsolved issues.

5.1.1 Parkinson's Disease Symptom Rating

Commonly, patients with PD show several characteristic movement dysfunctions: brady-hypokinesia, rigidity, tremor, postural instability, and movement initiation disorder ("freezing") [126–128]. Brady-hypokinesia is the most salient symptom of PD and is characterized by slowness and reduction of movements [129]. Rigidity is a stiffening of the body parts, clinically observed during passive movements [130]. Tremor typically occurs as a rest tremor, and is defined as an involuntary rhythmical muscle contraction with a frequency of 4-6 Hz [131].

The Movement Disorder Society sponsored Unified PD Rating Scale (MDS-UPDRS), was developed to assess the characteristic symptoms of patients with PD [62, 132]. The MDS-UPDRS is a 5-level rating scale and assesses motor symptoms, among others. It relies on patient-reported data and rater assessments. Commonly, clinical trials on neurorehabilative measures use the motor part of the MDS-UPDRS to establish clinical efficacy of any given intervention ([133–138]). The typical motor symptoms resulting from progressed PD can be grouped into three categories: bradykinesia with tremor, bradykinesia without tremor and dyskinesia. In this paper we refer to bradykinesia without tremor simply as *bradykinesia* and bradykinesia with tremor as *tremor*. We employ the MDS-UPDRS for rating the symptom severity.

5.1.2 Related Work and Open Problems

Numerous supervised machine learning techniques are available in literature to detect or estimate the unbalanced motor states and/or motor system abnormalities. We provide an overview of the relevant subset that addresses autonomous Parkinson's symptom detection and recognition in the following. Machine learning is employed in many studies to obtain relationships between sensor-based data and motion patterns related to movement dysfunctions of PD [133, 139–143]. According to a review of Kubota et al. [144], most approaches focus on only one or two symptoms out of the set of motor abnormalities *tremor, bradykinesia, dyskinesia*, and *gait disturbance*. However, our presented human motor model (see Chapter 2) allows for a comprehensive solution for PD motor symptom detection.

We review common methods for data acquisition and training data labeling, since supervised machine learning requires labeled data for the learning procedure. Thereafter, we introduce related work on autonomous PD motor symptom detection and estimation. We classify the approaches in the ones relying on linear versus nonlinear approximations.

Training Data Acquisition

As labeled training data is necessary for supervised machine learning, a movement disorder specialist commonly monitors the patients and labels clinical observations according to the MDS-UPDRS rating scale for PD. In the majority of such studies (e.g. [133, 139, 141, 145]) inertial measurement unit (IMU) based sensors attached to the patient's body were used. As an alternative to wearable sensors, video-based methods have been used to investigate the PD motor symptoms. Butt et al. [134, 143], for example, used the hand motion data collected using an RGB-D camera to recognize PD patients from healthy individuals with an accuracy of up to 85% using SVM.

In general, movements of the patients are commonly recorded using wearable sensors such as accelerometer, gyroscope, electromyography and video-based tracking devices such as infrared cameras. Marker-based motion tracking systems, in contrast, are considered as ground truth for other sensors [144, 146]. To ensure unsupervised usability by patients, only wearable inertial modules (accerlerometer and gyroscope) can be taken into consideration, as only those allow for prompt usage without significant impairment of the patients' quality of life.

Simple Approximations

Eskofier et al. [139] compare several supervised machine learning and deep learning algorithms to detect bradykinesia. Several specific motor tasks of 10 patients with idiopathic PD were recorded using IMU sensors. Each task was rated by a movement disorder specialist according to the MDS-UPDRS rating scale. Using these data, a classification accuracy of up to 85% was achieved with standard machine learning techniques such as SVM while deep learning demonstrated 90% accuracy in predicting the presence and absence of bradykinesia. Angeles et al. [133] were also able to classify PD symptoms such as kinetic tremor according to the MDS-UPDRS score given by clinicians with an accuracy of up to 87% using simple tree, linear SVM and k-nearest neighbor (kNN) algorithms. An accuracy of up to 92% was achieved in predicting bradykinesia using kNN.

The experiments for estimating motor aspects of PD described so far take place in laboratory environment, where the patients are asked to perform standardized activities. Few examples are available on symptom estimation during free living in the literature: For instance, in [147] binary classification of bradykinesia and tremor was achieved with 60-71% sensitivity, but differentiating the motor states ON and OFF yielded statistical significance for tremor only. Keijsers et al. [148] achieved an average sensitivity and specificity of 97% for binary classification in the motor states ON and OFF using 6 triaxial accelerometer modules on the patients' bodies. In contrast, a single waist sensor was used in a long-term (1-3 days) monitoring experiment by [149]. The average specificity and sensitivity were higher than 90% in the automatic assessment of patient's ON and OFF motor states during daily living. Semi-supervised classification algorithm based on k-means and self-organizing tree map clustering was applied in [150], obtaining accuracies in the range 42-99% for patients with different levels of dyskinesia severity.

The approaches presented so far allow for binary classification, but not for regression on symptom severity. Hence, the successful application scope for linear approaches is limited. Additionally, most approaches are not suitable for application in unconstrained daily living activities, since in general no straight forward generalization to non-laboratory environment is available.

Nonlinear Approximations

To estimate the symptom severity in contrast to a binary classification whether a symptom is present or not, requires more advanced machine learning techniques. Most prominent among the appropriate approaches for symptom severity recognition are GMM and GP regression. Such approaches, however, are rare in the literature. The system called PERFORM [151] uses accelerometer and gyroscope signals for machine learning, using the patient's own diary rating as ground truth. The output results were compared against the diary entries and a classification accuracy of 87% was achieved in predicting the presence and absence of tremor using a HMM. Predicting the presence and absence of bradykinesia, a classification accuracy of approximately 75% was achieved using SVM. Hence, again in this approach, symptom classification was binary. Further, the technology is restricted by specific hardware requirements. For the presented results, the PERFORM systems required four triaxial accelerometer devices to be physically attached the PD patient's body at each extremity, one accelerometer/gyroscope sensor on the waist and one data acquisition unit. Supervised machine learning is used in [141] to classify PD during free movement and daily living. The authors Cancela et al. report 70% to 86% classification accuracy for predicting the severity according to MDS-UPDRS, depending on the machine learning algorithm in free movement tasks. However, they only consider the symptom bradykinesia.

5.2 Data Collection

In this section we describe how the patient data was collected, introducing the patient cohort of our study, the sensor device and the setting for data capturing. Afterwards, the data acquisition itself is explained, which was performed during unrestricted daily living activities.

The individuals taking part in the study (30 in total) were diagnosed with PD by a neurologist according to UK Brain Bank Diagnostic Criteria [152] at the Schön Klinik München Schwabing, Germany. The average age of the participants was 67±10 and 20 were male and 10 were female. The mean disease duration was 11±5 years. The median of the patients' disease progress according to the Hoehn and Yahr scale [126] is 3.5 with an interquartile range of 1. The recruitment of the patient cohort and the data acquisition was performed at the Schön Klinik München Schwabing (Munich, Germany). This study was approved by the ethical board of the Technical University of Munich (Ref. No. 234/16S).

In order to learn and predict the PD symptoms using GP, the movements of the participants was recorded, together with corresponding clinical and activity information. The linear acceleration and angular velocity of the wrist were measured using the Microsoft Band 2 (Microsoft). Figure 5.1 illustrates how the device is worn. Inside the band is a 6-axis gyroscope/accelerometer module (LSM6D series by STMicroelectronics) and a bluetooth communication module (Bluetooth 4.0) for transmitting the data to a peripheral device. The accelerometer registers motions up to $\pm 8\ G$ ($G = 9.81\ m/s^2$) with a resolution of $0.244\ mG/LSB$ (least significant byte). The data range of the gyroscope is $\pm 1000\ dps$ (degree per second) with a resolution of 35 $mdps/LSB$. An Android application, "MS band data collector (pro)" was used to stream and store the data on a Samsung Galaxy A5 (Android 6.0.1).

Clinical observation of PD symptom and severity was concurrently performed by a trained expert who passively monitored the participants every minute during the data collection period. The motor symptom was described using the MDS-UPDRS, a standard 5-level rating scale, where bradykinesia corresponds to item III.14 and tremor to III.17. Dyskinesia was assessed using to the modified abnormal involuntary movement scale (AIMS, item A2.5). The balanced stage without abnormal motor symptoms is rated as 0 and the severity levels correspond to 1 = slight, 2 = mild, 3 = moderate and 4 = severe. Furthermore, voluntary

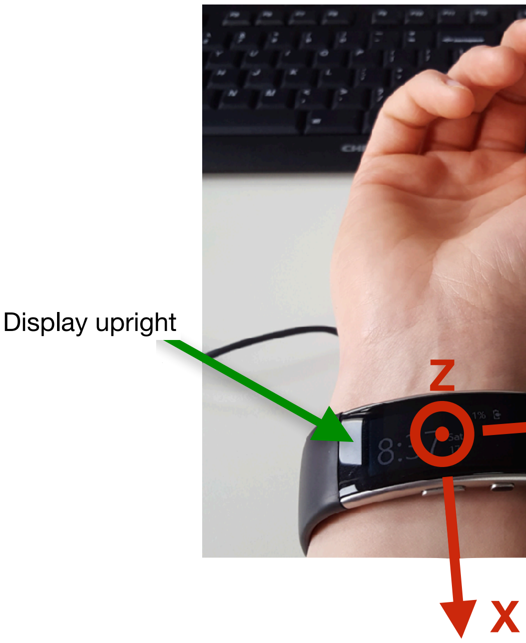


Figure 5.1: Wrist-worn inertial sensor Microsoft Band 2. The band is placed medially at the patient's wrist as displayed to assure the reference frames of all measurements to match.

activities (e.g., walking, standing, laying/resting and sitting) and medication intake (type and dosage) were reported in the same one-minute time window. When multiple symptoms and activities were present within the same time window, the predominant symptom and activity were reported.

The data collection was performed in a free-living environment during the regular in-patient stay at the hospital for drug dosage adaptation. On average, the data were collected for 331.2±192.6 minutes per participant, equaling to a total of 9937 minutes across all patients. After being briefed on the procedure, the participants wore the band on the wrist of the most affected side. Once the bluetooth connection was established between the band and smartphone which stored the data, the participants were free to engage in any daily routine including activities outside of the hospital. The recording ended when the patients desired, or before going to bed at the latest. Furthermore, the sensing device was disabled when the patients were in the toilet/bath or when requested.

5.3 PD Symptom Recognition

In this section, the approach for autonomous PD symptom detection and estimation is presented. We start by characterizing the collected data and their relevance for estimating PD symptoms, thereby motivating a selection of motion features. To consider the temporal evolution of PD symptom manifestation, we suggest to use the dynamical system approach for estimating the patients' motor symptoms introduced in Section 2.3. The PD symptom estimation (2.16) is then assessed by a multiple successive GP models: At first, a GP estimates the presence of tremor and its severity. In case tremor is absent, the following GP models are triggered, where two estimations are performed in parallel; one for dyskinesia and one for

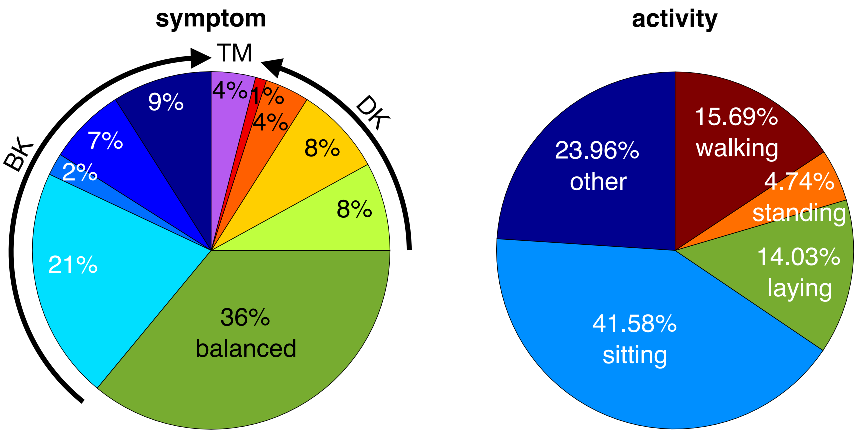


Figure 5.2: Distribution of symptom severities and activities of the participants. The label distributions were calculated from the collected dataset for each participant and then averaged. The arrows in the symptom chart (left) indicate increasing severity of dyskinesia (DK) and bradykinesia (BK) scores in 4 levels each. The tremor class (TM) as well has four severity levels, but is visualized here as a single section (purple).

bradykinesia without tremor. Hence, for all incoming data that passed the tremor estimation the severity level of both motor symptoms are predicted. The approach first estimates the tremor presence, in order to prevent the presence of tremor from impeding differentiation of bradykinesia with tremor to dyskinesia in inertia measurements. Then, we select the larger predicted value's symptom class as we intend to recognize the predominant motor symptom.

5.3.1 Data Analysis

In the following, we provide an overview of the collected patient data. The descriptive analysis of the symptom labels, collected by the clinical expert, shows that 35.95% belongs to the balanced class, while the bradykinesia and dyskinesia motor symptoms were observed in 38.70% and 21.13% of the data, respectively. The percentages are obtained by averaging the distributions of all participants. The labeled data also indicates that patients spent a large proportion of time sitting on a chair (41.58%), see Figure 5.2 for details. The activities gathered in the category *other* comprise specific tasks, for instance eating, climbing stairs and brushing teeth or activities with external acceleration like taking a train or elevator.

As different PD motor symptoms and patient activities result in a wide bandwidth of motion intensities, we investigate the signal power, as the power spectral density characterizes the frequency content of a signal. In (Figure 5.3) the power spectral density (PSD) of the accelerometer data is visualized. It shows that sitting and standing labels have PSD spread between $1-10$ Hz and a clearly differentiated tremor at around $4-6$ Hz PSD. For walking labels the characteristic tremor activity is partially absorbed by the walking frequency, which is at around 2.5 Hz in harmonics, as walking itself generates a strong PSD signal. Laying is generally described by low power across the spectrum, meaning that arm motion occurs infrequent. The shift of the tremor PSD peak towards 7 Hz during laying might be caused by the hand motion being constrained by a blanket. Moreover, the spectral analysis reveals on average increasing PSD in the symptom classes from bradykinesia through the balanced

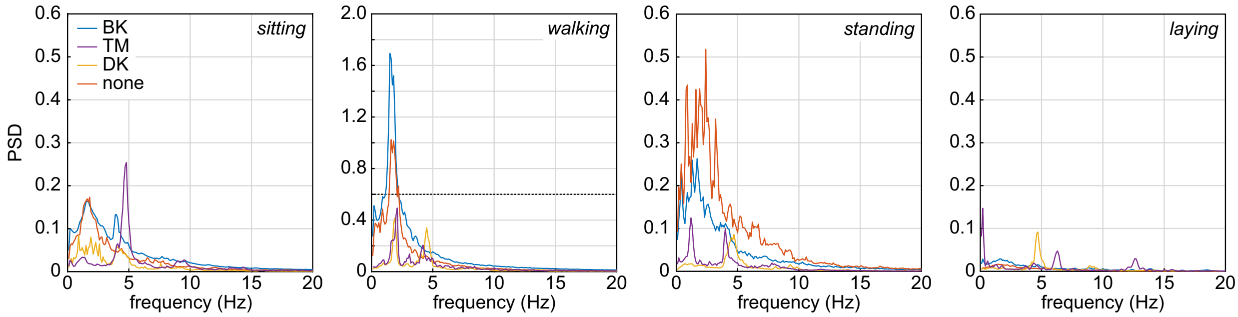


Figure 5.3: Power spectral density (PSD) of the accelerometer data. The PSD is visualized for bradykinesia (BK), tremor (TM), dyskinesia (DK) and the balanced condition (none). It shows generally high densities during walking, where the dotted line depicts the scale limit of the other three plots, and very low PSD values during laying.

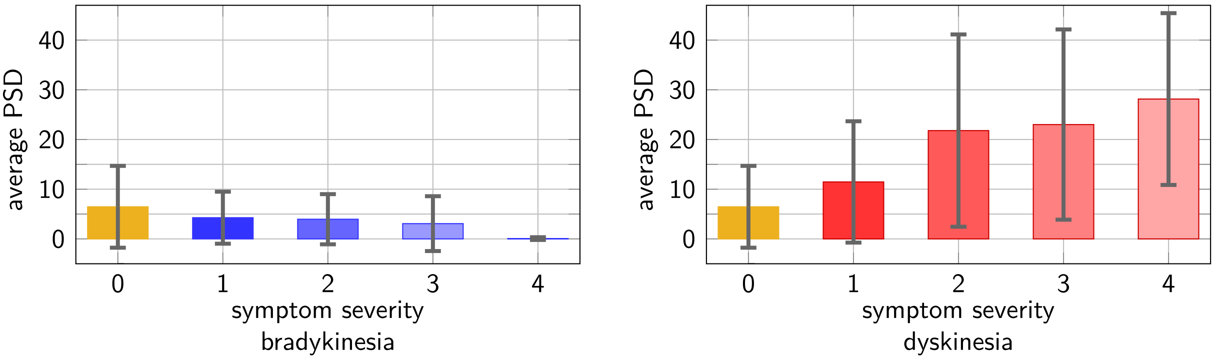


Figure 5.4: Average PSD of the non-tremor accelerometer data per symptom severity. The mean (visualized by bar height) of each PSD per symptom severity level with its standard deviation (gray error bar) is sorted from balanced to severe. In both figures the yellow bar depicts the average PSD level in the balanced condition. On the left side the symptom levels of bradykineasia (without tremor) are visualized in blue shades, on the right side the levels of dyskinesia in red shades.

condition to dyskinesia, as visualized in Figure 5.4, even though the individual activities introduce a high noise level. The tremor class is omitted in this figure, as the PSD of tremor data is dominated by the individual symptom manifestation of a patient.

The autocorrelation among symptom severity levels is investigated to assess the temporal relationship of PD symptoms. For each symptom class bradykinesia, dyskinesia and tremor the expert labels of the other classes are suppressed to zero and the respective symptom labels are normalized by the means and variances per patient. Then, the autocorrelation is calculated on the adapted labels for a lag of ± 15 min per patient. Figure 5.5 shows the average of the autocorrelation functions across patients. It indicates that the patient's motor symptoms can be modeled as a temporal evolution. A slow transition is present in the autocorrelation coefficients of the fluctuations in the symptom severity for dys- and bradykinsia, whereas the transition is comparably faster for tremor.

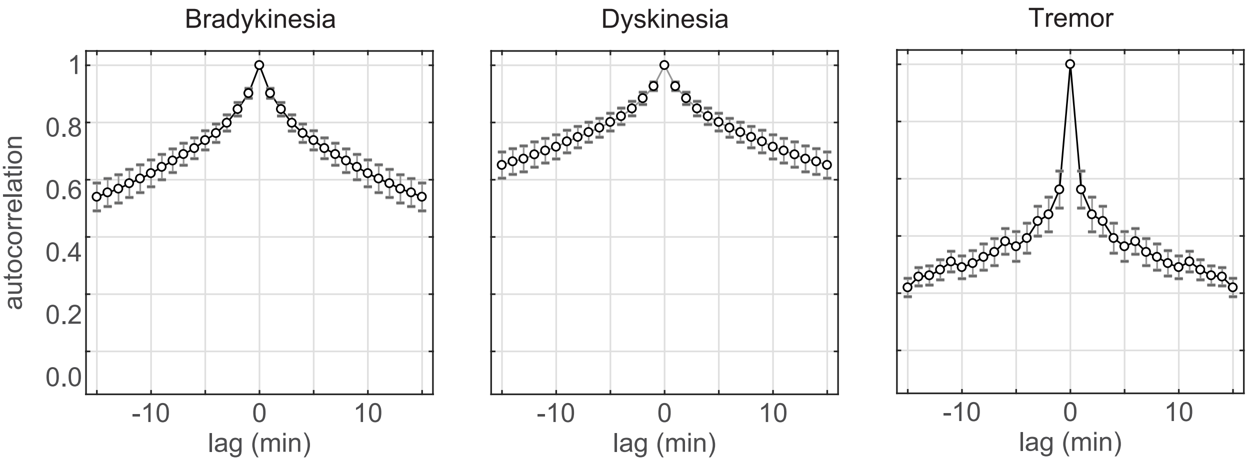


Figure 5.5: Autocorrelation of symptom labels. Autocorrelation coefficients of the bradykinesia, dyskinesia and tremor fluctuations indicate a very slow transition of dys- and bradykinesia severity whereas the transition is comparably faster for tremor severity. The data were averaged across the (available) patients, and the error bars indicate one standard deviation.

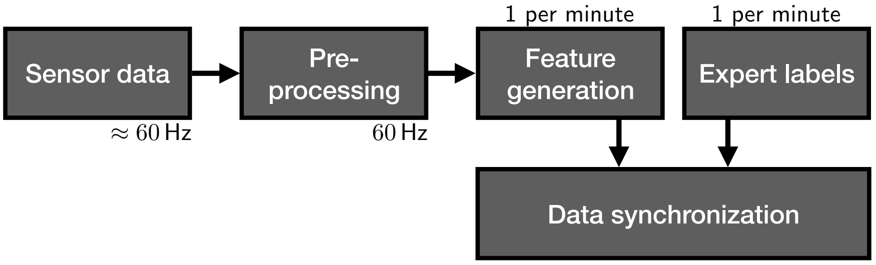


Figure 5.6: Inertial data and expert ratings processing. The sensor data is pre-processed by applying a two-directional Butterworth filter and wavelet decomposition, before clinically relevant PD features are extracted. The result is then synchronized with the activity and symptom severity labels of the expert rater.

5.3.2 Monitoring Unit: Data Processing and Feature Generation

The collected inertial data, consisting of accelerometer and gyroscope data, are processed to quantify relevant PD features, which are then synchronized with the expert ratings and activity labels as schematically visualized in Figure 5.6. The resulting data set is used for estimating the symptoms and severities in the fault detection unit.

As the raw inertial data includes sensor noise, we mildly filter the data by applying a two-directional Butterworth filter to accelerometer and gyroscope data, i.e. to the measured linear accelerations $\{(\alpha_x, \alpha_y, \alpha_z)_i^\top\}_{i=1}^{\nu}$ and rotational velocities $\{(\omega_\phi, \omega_\psi, \omega_\vartheta)_i^\top\}_{i=1}^{\nu}$, respectively.

$$\begin{aligned} (\tilde{\alpha}_x, \tilde{\alpha}_y, \tilde{\alpha}_z)^\top &= f_{\mathrm{Bw}}\left(\alpha_x, \alpha_y, \alpha_z, b_l, b_u\right), \\ (\tilde{\omega}_\phi, \tilde{\omega}_\psi, \tilde{\omega}_\vartheta)^\top &= f_{\mathrm{Bw}}\left(\omega_\phi, \omega_\psi, \omega_\vartheta, b_l, b_u\right). \end{aligned} \tag{5.1}$$

The lower bound b_l of the cut-off frequency was set to 0.1 Hz to filter out sensor drift, and the upper bound b_u was set to 20 Hz to filter out high frequency noise. Furthermore, to avoid dependency of the signal on the wrist band placement (left versus right wrist) and orientation (lateral versus distal and upright versus inverted), the vector norm $\| \cdot \|$ of both filtered inertial units is calculated,

$$\begin{aligned} \mathfrak{s}_{\text{acc}} &= \| \, (\tilde{\alpha}_x, \tilde{\alpha}_y, \tilde{\alpha}_z)^\top \, \| \\ \mathfrak{s}_{\text{gyr}} &= \| \, (\tilde{\omega}_\phi, \tilde{\omega}_\psi, \tilde{\omega}_\vartheta)^\top \, \|, \end{aligned} \tag{5.2}$$

and thus scalar signals $\mathfrak{s}_{\text{acc}}$ and $\mathfrak{s}_{\text{gyr}}$ are obtained. All subsequent feature generation is performed on the processed signals.

As demonstrated in Figure 5.4, the different symptoms have different PSD characteristics. Therefore, we base the feature generation on a time-frequency transformation, namely on wavelet decomposition of the processed sensor data. The signals $\mathfrak{s}_{\text{acc}}$ and $\mathfrak{s}_{\text{gyr}}$ are transformed using Daubechies wavelets ϖ_3 of order 3. The odd-numbered decomposition level 1, 3, 5, 7 and 9 are employed, as those layers cover the bandwidth of activity levels present in daily living activities. In Figure 5.7 a raw accelerometer signal and the third wavelet decomposed level of the corresponding filtered signal vector norm is depicted. The lower part of the figure shows that wavelet decomposition is capable of differentiating voluntary motion (white background) from the tremor symptom (green shaded area). We remove the even-numbered layers from the model to minimize redundancy in the feature space. Then, for each decomposed level $\tilde{\mathfrak{s}}_{\text{acc},i} = \varpi_3(\mathfrak{s}_{\text{acc}}, i)$ and $\tilde{\mathfrak{s}}_{\text{gyr},i} = \varpi_3(\mathfrak{s}_{\text{gyr}}, i)$, where $i \in \{1, 3, 5, 7, 9\}$, characteristic features are calculated. The features consist of standard deviation, norm, maximum, root mean square, kurtosis and skewness, as they encode motion properties of the previously presented PD motor symptoms and patient activities. In addition, the signals $\tilde{\mathfrak{s}}_{\text{acc},i}$ and $\tilde{\mathfrak{s}}_{\text{gyr},i}$ are differentiated for all $i \in \{1, 3, 5, 7, 9\}$ and the standard deviation, norm and root mean square are reapplied to the differentiated signals. A logarithmic scaling is used on a selection of features to improve the activity level separation, as the logarithmizing stretches small positive signals. More specifically, the logarithm is taken of all features obtained from the gyroscope and of the differentiated accelerometer features.

Every feature is computed for each one minute time window $\mathbf{t}$ corresponding to a label by the expert rater, if the size of the sample set $\mathcal{J}_{\mathbf{t}}$, that is captured during window $\mathbf{t}$, contains at least 10% of the number of data samples that should be captured during one minute, i.e. $\#\mathcal{J}_{\mathbf{t}} \geq 360$ sensor measurements (60 Hz sampling rate times 60 seconds). Hence, we only omit data windows that suffer from severe data loss. We allow this small percentage of data samples per captured time window, because the data transmission via Bluetooth between sensor device and storage device is unreliable due to external interferences and consequently frequent data losses are apparent.

The features introduced so far characterize the power in the inertial data and are thus best suitable to describe the symptoms tremor and dyskinesia. However, bradykinesia is characterized by very slow motion and entire absence of motions. Therefore, rest phases in the patients' sensor data additionally need to be quantified. Therefore, features encoding the amount of rest phases within the time window are investigated. We define rest as the proportion of data where the processed inertial signals (5.2) are below given thresholds $a > 0$,

$$\text{rest}_{\text{acc/gyr}} = \sum_{j \in \mathcal{J}_{\mathbf{t}}} \frac{\mathbb{B}\left(\mathfrak{s}_{\text{acc/gyr}}(j) < a\right)}{\#\mathcal{J}_{\mathbf{t}}}, \tag{5.3}$$

where $\mathbb{B}(\cdot)$ denotes the boolean operator that assigns the numbers $\{0, 1\}$ depending on whether the relation on the inside is false or true. Multiple thresholds are introduced to

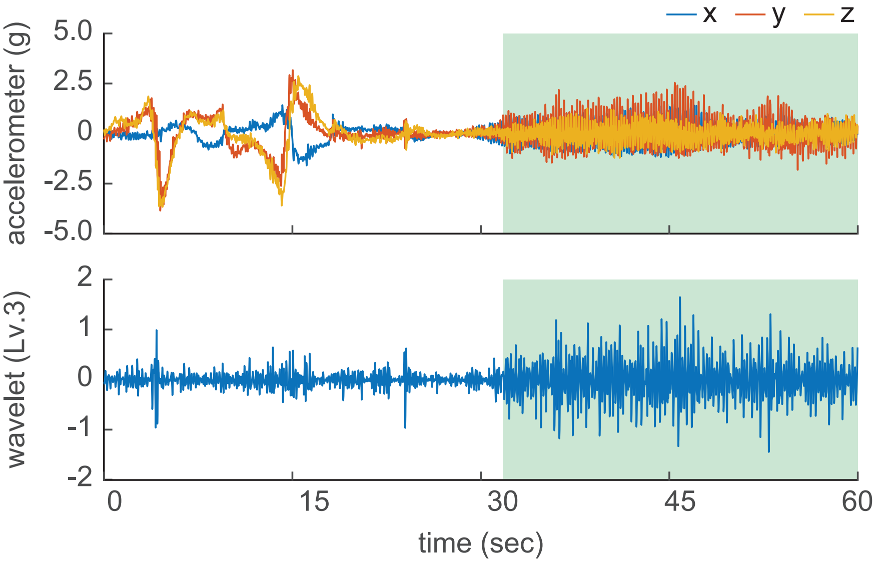


Figure 5.7: Wavelet transformation of the accelerometer data. The discrete wavelet decomposition is performed on the vector norm of 3D accelerometer (and gyroscope) measurements to calculate features (e.g. standard deviation, norm and root mean square) that describe characteristics of the 60 seconds time window. Only the third level of the accelerometer signal decomposition is depicted. The green shaded area indicates presence of the tremor label.

compensate for inter-patient and inter-activity variability and to cover all symptom severity levels. We use 0.1, 0.15, 0.2, 0.25 and 0.3 G as thresholds a for accelerometer data and 1, 1.25, 1.5, 1.75 and 2 dps for gyroscope data, respectively. As during severe bradykinesia it can happen that the patient does not move during the whole one minute window, additionally the rest proportion over a 5 minute window $\bigcup_{\mathbf{t}\in\mathcal{T}} \mathcal{J}_{\mathbf{t}}$ is calculated, consisting of the two minutes before and after the current time window, $\mathcal{T} = \{\mathbf{t}_{-2}, \mathbf{t}_{-1}, \mathbf{t}, \mathbf{t}_{+1}, \mathbf{t}_{+2}\}$. As autocorrelation of bradykinesia is high for long time intervals, see Figure 5.5, the overall condition of the patient is assumed to be the same during those 5 minute windows.

Additionally to the previously mentioned features, we include features that are inspired by the Parkinson's KinetiGraph system [153]. The two raw inertial signals are filtered, using a stronger bandpass filter (with limits 0.2 Hz and 4 Hz) to keep voluntary motion only. Then, the maximum and the mean spectral power at the maximum are calculated, as those features have been reported to provide promising results for PD symptom detection [154].

The total number of features obtained is 132 for both inertial sensors (the accelerometer and gyroscope) together. They are joined into the feature vector $\mathbf{z} \in \mathbb{R}^{132}$. It encodes the clinically relevant characteristics of the motor symptoms tremor, bradykinesia and dyskinesia. We use this vector in our multi-layer approach for autonomous symptom recognition.

Remark 5.3.1 *Through the data processing and feature generation, which we presented, we transform the captured executed motor behavior into a feature vector* $\mathbf{z}_\kappa$ *that incorporates important features that characterize motor dysfunctions as well as intended motion. The frequency analysis allows us to determine the amount of those characteristics in the captured signal. Hence, the feature vector is a suitable basis to conduct fault detection on.*

5.3.3 Fault Detection Unit: Multi-layer GP Model

In this section, we present our technique to realize the fault detection (2.16), which we introduced in Section 2.3. We consider as given time wise synchronized data sets for patients' symptom severity assessment $\{\xi_\kappa\}_{\kappa=0}^{\nu}$ and processed inertial sensor measurements $\{\,\mathbf{z}_\kappa\}_{\kappa=1}^{\nu}$ from a wrist-worn wearable sensor. To realize the estimation (2.16), we suggest to use multiple successive GPs with the squared exponential kernel, since those GPs are well suited for approximating human movement behavior due to their property to generate smooth motion predictions. We call the procedure of applying multiple GP models successively a *multi-layer GP model.*

The employed training input $\{\,\mathbf{x}_\kappa\}_{\kappa=1}^{\nu}$ to the multi-layer GP consists of the previous symptom levels of each patient $\{\xi_\kappa\}_{\kappa=0}^{\nu-1}$ and the feature vector $\{\,\mathbf{z}_\kappa\}_{\kappa=1}^{\nu}$ appended to the feature vector,

$$\mathbf{x}_\kappa = (\,\mathbf{z}_\kappa, \xi_{\kappa-1})^\top \text{ for training.} \tag{5.4}$$

The process output describes a Gaussian distribution, which has the GP mean prediction as expected value $\mathbb{E}[\cdot]$. Hence, we consider in the following only the GP mean predictions as process output y_κ, as the mean describes an estimate for the κ-th clinical symptom severity rating ξ_κ.

Specifically, at first a **tremor GP** is trained to recognize the presence and severity of tremor from the training input set $\left\{(\,\mathbf{z}_\kappa, \xi_{\kappa-1})^\top\right\}_{\kappa=1}^{\nu}$. The training output set $\{y_{\mathsf{TM},\kappa}\}_{\kappa=1}^{\nu}$ is optimized to approximate the symptom,

$$y_{\mathsf{TM},\kappa} \approx \begin{cases} \xi_\kappa & \text{if } k \in \mathcal{J}_{\mathsf{TM}}, \\ 0 & \text{otherwise,} \end{cases} \tag{5.5}$$

where the set $\mathcal{J}_{\mathsf{TM}} := \{j \in \{1, \dots, \nu\} \wedge \xi_j \text{ contains tremor}\}$. The next layer is triggered for the non-tremor data only in the measurement set $\left\{(\,\mathbf{z}_\kappa, \xi_{\kappa-1})^\top\right\}_{\kappa=1, \kappa\notin\mathcal{J}_{\mathsf{TM}}}^{\nu}$. This second layer comprises two GP estimations; the **dyskinesia GP** for modeling dyskinesia, the **bradykinesia GP** for modeling bradykinesia. Each of the GPs is trained to approximate the symptom and severity with its output

$$\begin{aligned} y_{\mathsf{DK},\kappa} &\approx \begin{cases} \xi_\kappa & \text{if } \kappa \in \mathcal{J}_{\mathsf{DK}} \wedge \kappa \notin \mathcal{J}_{\mathsf{TM}}, \\ 0 & \text{else,} \end{cases} \\ y_{\mathsf{BK},\kappa} &\approx \begin{cases} \xi_\kappa & \text{if } \kappa \in \mathcal{J}_{\mathsf{BK}} \wedge \kappa \notin \mathcal{J}_{\mathsf{TM}}, \\ 0 & \text{else,} \end{cases} \end{aligned} \tag{5.6}$$

where the index sets $\mathcal{J}_{\mathsf{DK}}$ and $\mathcal{J}_{\mathsf{BK}}$ are defined as $\{j \in \{1, \dots, \nu\} \wedge \xi_j \text{ dyskinetic}\}$ and $\{j \in \{1, \dots, \nu\} \wedge \xi_j \text{ bradykinetic}\}$, respectively. In the third layer the decision among the balanced, dyskinesia and bradykinesia classes is made based on the results from layer two: When both GP models provide outputs $y_{\mathsf{DK},\kappa} < \tilde{a}$ and $y_{\mathsf{BK},\kappa} < \tilde{a}$ below a certain threshold $\tilde{a}$, we consider the correct classification to be balanced. Otherwise the symptom class of the GP model providing the higher predicted value is selected.

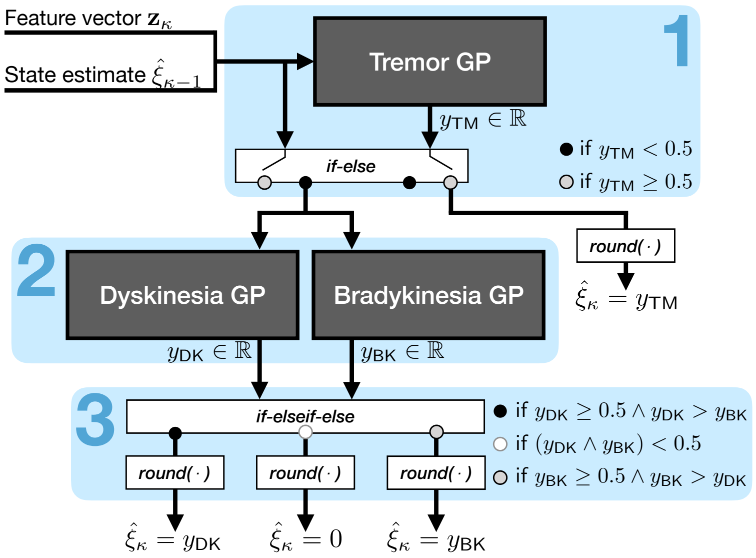


Figure 5.8: Scheme of multi-layer algorithm. The feature vector $\mathbf{z}_\kappa$ and the state estimate $\xi_{\kappa-1}$ are the inputs to the multi-layer approach. After testing for tremor in the first model layer ($y_{\mathrm{TM}} < 0.5$ or $y_{\mathrm{TM}} \geq 0.5$), the remaining non-tremor data is tested in the second layer for dyskinesia and bradykinesia severity. Each of the GPs provides an estimate of the symptom severity in the continuous space $\mathbb{R}$, where, however, values outside the interval $[-0.5, 4.49]$ are extremely rare. In the third layer of the hierachical approach we decide among the three remaining motor symptom conditions balanced, bradykinesia and dyskinesia depending on the estimated symptom severities. If both predictions obtained from the dyskinesia GP and the bradykinesia GP, respectively, are below a certain threshold ($\tilde{a} = 0.5$), the balanced condition is nominated. If any of the predicted values exceeds the threshold, the symptom class of the larger value is appointed. To determine one of the categorical symptom severity levels $1-4$, the GP prediction is rounded to an integer value.

Remark 5.3.2 *This multi-layer GP structure implicitly comprises two types of information about the motor symptom: It tells us, which motor dysfunction is present in the motor behavior and how severe the corresponding symptom is.*

After the model training is finished in the multi-layer GP, we aim to provide symptom estimates for unseen input data. As, however, the available measurements only consist of the feature vector $\mathbf{z}^*_\kappa \in \mathbb{R}^{132}, \kappa > 0$, we initialize the internal symptom severity $\hat{\xi}^*_0 \in \{0, \dots, 4\}$ by generating a uniformly drawn random integer from $[0, 4]$. Thus, we obtain as GP input

$$\mathbf{x}^*_\kappa = \left(\mathbf{z}^*_\kappa, \hat{\xi}^*_{\kappa-1}\right)^\top \text{ for testing,} \tag{5.7}$$

where the previously predicted output $y^*_{\kappa-1}$ is used for the symptom level estimate $\hat{\xi}^*_{\kappa-1} = \lfloor y^*_{\kappa-1} \rceil$. The function $\lfloor \cdot \rceil$ rounds the GP mean prediction $y^*_\kappa \in \mathbb{R}$, which is obtained in continuous space, to the nearest integer, and in the unlikely case of predictions outside the interval $[-0.5, 4.5)$, maps the negative and positive values to 0 and 4, respectively. It is required to meet the categorical format $\{0, 1, 2, 3, 4\}$ of the standard PD rating scale. The individual steps inside the algorithm are the following. The tremor GP provides in the first layer a symptom severity estimate y_{TM} for the new input $\mathbf{x}^*_\kappa$. If no tremor is indicated, predictions y_{DK} and y_{BK} for the severity of dyskinesia and bradykinesia, respectively, are provided in the second layer, and the decision for one of the symptom classes is made in the third layer. An illustration of the multi-layer approach, estimating the measurable system output (2.16), is provided in Figure 5.8.

Remark 5.3.3 *The fault detection estimates the symptom type and the severity level depending on the feature vector and the previously estimated motor symptom, see Section 2.3. Hence, we predict the symptom severity model-based through the function g_f inside* (2.16), *where the feature vector $\mathbf{z}_\kappa$ contributes to the measurement-based innovation. The fault detection is realized through employing multiple successive GP models that are arranged in a multi-layer GP model. Hence, when we managed the free parameters in the multi-layer GP, we are ready to apply the estimation to real human data.*

5.3.4 GP Hyperparameter Characteristics

To ensure that the GPs generalize well to unseen data, the GPs should learn the symptom characteristics instead of the data distribution. Therefore, the symptom estimation false positives (*FP*, i.e. predicting a symptom where there is none) and false negatives (*FN*, i.e. missing the presence of a symptom) are considered as equally undesired, as this induces impediment of over- and underestimation of symptoms at the same time. This means the GP models in Figure 5.8 are required to satisfy the property $FN/FP = 1$, which is achieved by selecting the initial hyperparameters of each of the GP models to approximate this ratio. As a GP model that in the initial optimization step roughly meets the $FN/FP = 1$ property is already close to a local optimum and the GP model training employs a gradient descend algorithm, it is unlikely that during training the model deviates from producing estimates where $FN/FP \approx 1$. Further, a relatively large signal noise ϑ_n hinders the overfitting of the GP models to the training data sets and thus, reduces the model's training accuracy, but during testing supports the generalization property.

5.4 Statistical Analysis

In this section we detail the practical application of the multi-layer approach to the patient data set. First, we introduce a feature vector reduction, to ensure efficient performance of the multi-layer GP. Then, we present how the patient data set is split into disjoint training and test sets, to analyze the multi-layer model's ability to generalize to unseen data and unknown patients. Further, we explain how we deal with the non-uniform symptom distribution in the patient set. Finally, we provide the initial training hyperparamters and independent training accuracies of the tremor GP, the dyskinesia GP and the bradykinesia GP.

5.4.1 Feature Vector Reduction

To reduce the computational complexity of the multi-layer GP, each internal GP model is trained and tested on a subset of the 132 dimensional feature vector only. To determine the informative wavelet decomposition levels a detailed interpretation of the behavioral motion spectrum of the patient data is required: Decomposition level 1 contains motions with very low frequency (<0.2 Hz) which is slower than usual human behavior and thus can be assumed to mainly contain sensor drift. Decomposition levels 3 and 5 cover the frequency range of most intended motions and are thus important to distinguish voluntary motion from PD symptoms. Decomposition level 7 includes frequencies that correspond to fast movements, rarely found in the voluntary motion spectrum of daily living activities of elderly PD patients, but frequently occurring during dyskinetic symptom phases. Decomposition level 9 contains motions of the characteristic tremor frequencies ($4 - 6$ Hz).

Therefore, we reduce the feature vector $\mathbf{z} \in \mathbb{R}^{132}$ for the dyskinesia GP about the wavelet decomposition levels 1 and 9 for both signals (accelerometer and gyroscope), and for the bradykinesia GP about the levels 1 and 7 for both signals. Hence, in each of the GP models the dimensionality of the input vector is reduced about $36 = 2 \times 2 \times 9$ dimensions (9 is the number of features calculated in one wavelet decomposition level). In the tremor GP the full feature vector $\mathbf{z} \in \mathbb{R}^{132}$ is employed to facilitate separation of the symptom from all other incoming signals.

5.4.2 Disjointed Patient Sets

We split the patient cohort, consisting of 30 participants, into two disjoint sets; one group consists of the training patients, whose data is used to train the multi-layer approach, the other group is the test patients, whose so far unseen data is used to test the trained model's accuracy. With this procedure, we not only quantify the ability of our approach for perform regression and prediction, but moreover demonstrate the model's ability to generalize to unknown participants. Hence, we introduce an approach that does not require fine tuning on the target patient, but is globally applicable to PD patients.

Specifically, we perform a leave-one-subject-out (LOSO) approach, where we repeat the training and testing procedure 30 times, in each of the independent runs the test group consists of one participant and the training is performed on the remaining 29 patients in the cohort. We iterate trough the patient cohort so that after the 30 trials every patient was once the test patient.

5.4.3 Non-uniform Symptom Distribution

With respect to the three independent rating scales for the symptoms bradykinesia, dyskinesia and tremor, the PD data set, is extremely non-uniform distributed. In Figure 5.9, top row, the data distributions according to the independent scales are visualized. For instance, according to the tremor rating scale, 96% of the data does not show this symptom and thus, each of the 30 tremor GPs has an underlying symptom distribution similar to the pie chart in the upper right corner of Figure 5.9. We use "similar", because in each LOSO experiment run a different patient is left out, which effects slightly the symptom distribution of the training data set of the remaining 29 patients, whereas Figure 5.9 shows the symptom distribution of the full data set.

Finding suitable GP model hyperparameters that represent the symptom characteristics, not the symptom frequency, gets more difficult the more non-uniformly the symptoms are distributed in the training data set. Therefore, we do not train the three GP models on the full data set, but on data subsets that only comprise the balanced data and the data

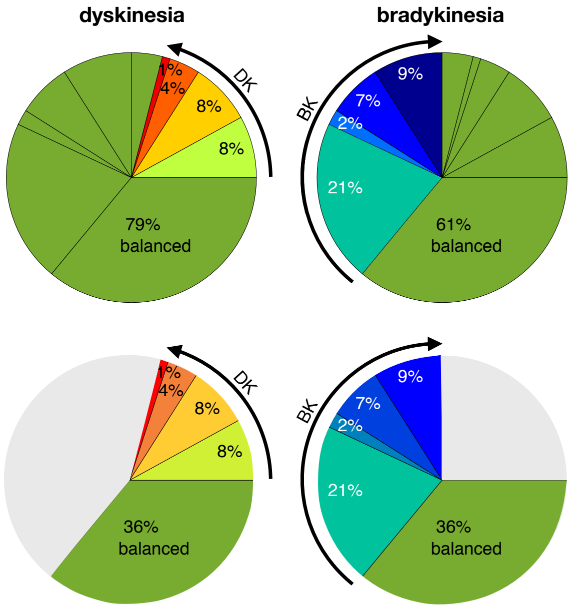


Figure 5.9: Distribution of symptom severities according to the different rating scales. According to the independent scales all data where a specific symptom is not apparent is assigned to the balanced class. To reduce the non-uniformity of the symptom distribution during the GP model training, the dyskinesia GP, the bradykinesia GP and the tremor GP are trained on the data subsets illustrated in the bottom row left, middle and right, respectively. For all three GPs the model testing, however, is performed on the full data sets (top row).

where the respective symptom (bradykinesia, dyskinesia or tremor) is present, see Figure 5.9, bottom row. We do not further reduce the amount of balanced data, as the balanced class covers the most widespread activities in free living and thus, contains a large diversity of intended motion patterns that need to be distinguished from the unintended motions caused by the symptoms.

The model testing, however, is performed on the full data set, i.e. the GP predictions of each of the GP models in Figure 5.8 are tested against the expert labels of the respective test patient regardless of the symptom class.

5.4.4 Initial Hyperparameters

All 30 training runs per GP model class (tremor, dyskinesia and bradykinesia) are initialized with the same vector of starting hyperparameters $\mathbf{h} = (\lambda, \sigma_f, \sigma_n)^\top$, in particular for any tremor GP the initial hyperparamter values are $\mathbf{h}_{\mathsf{TM}} = (96.83, 0.23, 0.50)$, for any bradykinesia GP they are $\mathbf{h}_{\mathsf{BK}} = (96302550, 826659, 0.65)$ and for any dyskinesia GP they are $\mathbf{h}_{\mathsf{DK}} = (10572, 148, 0.83)$. Those hyperparameter vectors, found by heuristics, fulfill the $FN/FP \approx 1$ property.

5.4.5 Training Accuracy of Symptom Severity

The three GP model classes in the multi-layer approach are trained independently using the described setup for GP model training. We denote with *training accuracy* the percentage

Table 5.1: Training accuracy of the symptom severity for each activity.

Symptom	accuracy (%)	other	sitting	walking	standing	laying	all
tremor	mean	89.06	79.75	82.81	78.11	86.03	83.15
	std	1.24	1.58	2.05	2.83	1.66	1.87
bradykinesia	mean	66.39	61.05	62.03	57.50	62.72	61.94
	std	5.70	6.44	6.02	7.03	6.00	6.24
dyskinesia	mean	51.72	58.98	51.67	56.35	81.25	59.99
	std	3.23	2.20	2.27	4.36	1.22	2.66

of data, where (after the model training has finished) the re-estimated symptom severity matches the expert label, and provide the achieved model training accuracies in Table 5.1. Row-wise the training accuracy of the tremor GP, dyskinesia GP and bradykinesia GP is provided in terms of the mean and the standard deviation of the 30 independent experiment runs in the LOSO approach. The severity estimation accuracy percentages are given for each of the main activity categories (*sitting, walking, standing, laying* and *other*) and for the total amount of data under *all.* The presented accuracies in all GP models are normalized by the amount of data available in each run, to prevent biased results due to the distortion of repetitions, where the data set for a certain symptom class differs strongly in size.

5.5 Results

In this section we report the experimental results of our approach. We present the multi-layer GP test accuracies, where we first regard each model layer as a standalone task and report the accuracy of each GP model independently, before we provide the total accuracy probabilities of the second layer, conditioned on the accuracy of layer one. All GP model accuracy results are provided in terms of the mean and standard deviation per main activity category (*sitting, laying, standing, walking* and *other*). For comparison, we additionally present the results for pure regression where the GP input for both training and testing is $\mathbf{x}_\kappa = \mathbf{z}_\kappa$, i.e. without considering a dynamical system in the approximation.

5.5.1 Individual and Total GP Model Accuracies

In the following, the test accuracies in predicting the patient's symptom and severity level of the three GP classes in the multi-layer approach are provided. Besides the *standard accuracy*, the percentage of predictions where the expert labels $0-4$ match the model output, we provide the ± 1 *accuracy*, defining the percentage when the predicted symptom severity is at most one level off the severity assessed by the expert rater. The prediction accuracy of the full test data is provided in column *all*, while the accuracies for each of the activity categories *sitting, walking, standing, laying* and *other* are presented in the corresponding column. The results are presented in terms of mean and standard deviation (std) of the accuracies in each of the conducted 30 test runs when iterating through the test patients. The accuracy results are normalized by the amount of data available for each patient and activity to avoid any bias in calculating the mean and standard deviation.

We start by analyzing the accuracies of each layer independently. Hence, the accuracies in the second layer are obtained assuming 100% correct tremor detection in the first layer. In Table 5.2 the independent in-layer results are presented. The symptom severity estimations exceeds 75% accuracy in each activity for the tremor GP and in less than 5% of cases is the predicted symptom severity more than one level off the expert ratings for each activity. Hence, the tremor estimation in layer one is soundly separating the patient data into tremor and non-tremor data, and provides quite accurate predictions of the tremor severity independent of the performed activity. The captured movements' composition of intended motion and the unintended motor symptoms of dyskinesia and bradykinesia varies from patient to patient, and usually requires individual model tuning to the test patient. Our approach of training the GP models with unified initial hyperparameters is designed to generalize among patients. Therefore, it shows in cases a high standard deviation in the prediction accuracy, indicating a decreased suitability of the GP models for individual patients with atypical movement composition in comparison to the training patient set. However, the accuracy of symptom severity estimations in layer two never falls below 62% and 50% for bradykinesia and dyskinesia, respectively. In some activities the motor dysfunctions dominate more than in others. Therefore, the dyskinesia GP performs particularly well in estimating symptoms during laying, while the most difficult activity for the bradykinesia GP is standing. The ± 1 accuracy of the severity estimation for both motor symptoms clearly exceeds 80% through all activities except for the bradykinesia GP during laying. Laying is frequently reported for patients where severe bradykinesia symptoms are observed. The severity predictions of the dyskinesia GP during laying, however, are in more than 99% correct or by at most 1 level off. This activity generally is disturbed by few intended motions and thus, shows pure motor symptoms often. For the testing results we obtain averaged FN/FP ratios of 1.16, 0.99 and 0.99 for the tremor GP, the bradykinesia GP and the dyskinesia GP, respectively.

Next, we investigate the *total accuracy* of the symptom predictions, i.e. the probability of both layers (the first and the second) being predicted correctly at the same time. Specifically, the total accuracies are the percentage of accurate predictions of the bradykinesia GP and the dyskinesia GP, respectively, intersected with the accuracy of the tremor GP. For simplicity of presentation we provide the accuracy and the ± 1 accuracy in terms of the percentage of correct predictions of the multi-layer approach for all patients taken together. In total, in 1318 instances of GP predictions tremor is estimated. The remaining 8619 data samples are processed in the second layer of the multi-layer approach. Inside the second layer, the bradykinesia GP and the dyskinesia GP never falsely predicted the presence of both symptoms in parallel for data of the same one minute time window. Hence, symptom discrimination worked precisely with the proposed motion features. Table 5.3 notes the total test accuracy for the PD symptoms tremor, bradykinesia and dyskinesia and for the balanced condition, where movement disorders are absent. The accuracies of the severity estimations $(0-4)$ are provided for the activity categories sitting, walking, standing, laying, other and all. For the full data set the tremor severity is estimated accurately in more than 80%, the bradykinesia and dyskinesia severity in more than 50% and the balanced condition is detected correctly in 48%. For all motor symptoms the ± 1 severity estimation accuracy is above 80%, demonstrating a reliable motor symptom detection tendency.

In the above experiments we trained and tested on the dynamics (2.16) with input (5.4) and (5.7), respectively. For comparison we present in the following results for pure regression

Table 5.2: Individual layer testing accuracy of the symptom severity for each activity.

Symptom	accuracy (%)	other	sitting	walking	standing	laying	all
tremor	mean	88.77	79.13	81.56	78.50	84.97	82.58
	std	15.72	24.37	20.58	26.53	15.64	20.57
	±1 mean	98.92	95.53	97.77	95.74	97.52	97.09
	±1 std	5.17	9.22	7.57	11.93	6.55	8.09
bradykinesia	mean	71.40	71.91	72.97	62.70	67.64	69.32
	std	31.54	33.25	35.63	33.46	39.09	4.59
	±1 90.24	91.97	92.40	94.85	79.24	89.74	
	±1 std	17.08	15.97	15.06	9.54	34.99	18.52
dyskinesia	mean	51.09	56.87	50.00	50.70	86.21	58.97
	std	18.53	20.37	21.48	25.07	9.85	19.06
	±1 mean	89.19	90.03	88.32	87.99	99.19	90.94
	±1 std	21.78	15.71	15.09	15.53	2.37	14.10

Table 5.3: Total testing accuracy of the symptom severity for each activity.

Motor symptom	severity (%)	other	sitting	walking	standing	laying	all
tremor	accuracy	88.77	79.13	81.56	78.50	84.97	82.58
	±1 accuracy	98.92	95.53	97.77	95.74	97.52	97.50
bradykinesia	accuracy	67.48	67.47	63.37	46.32	60.38	64.68
	±1 accuracy	84.66	85.77	80.36	86.76	72.93	82.76
dyskinesia	accuracy	45.77	51.84	42.76	43.38	81.29	53.20
	±1 accuracy	83.96	83.00	76.70	79.66	90.76	83.18
balanced	accuracy	45.47	53.16	36.86	35.34	72.77	48.56
	±1 accuracy	90.14	87.70	77.70	80.42	93.41	85.64

without a dynamical system in the approach, i.e. where the GP input for both training and testing equals the feature vector $\mathbf{x}_\kappa = \mathbf{z}_\kappa$. The accuracy results for the bradykinesia GP and dyskinesia GP are provided in Table 5.4, as for those symptoms the autocorrelation is high (as shown in Fig 5.5), and consequently, considering a dynamical system significantly influences the results. The severity prediction accuracy significantly drops in each activity for both GP models in comparison to the results including the dynamics in the model (see Table 5.2).

Table 5.4: Individual layer testing accuracy of the symptom severity for each activity without dynamics.

GP type	accuracy (%)	other	sitting	walking	standing	laying	all
bradykinesia	mean	52.05	53.82	55.70	48.28	33.90	48.75
	std	12.85	14.35	14.53	17.22	11.61	14.11
	±1 mean	89.10	90.17	91.83	89.22	77.11	87.49
	±1 std	6.91	10.72	8.16	10.77	18.50	11.01
dyskinesia	mean	42.28	48.74	41.76	44.94	81.15	51.77
	std	29.67	26.14	31.71	31.13	17.09	27.15
	±1 mean	87.46	89.02	91.04	88.63	93.36	89.90
	±1 std	23.45	18.47	16.21	22.58	6.58	17.46

5.5.2 Interpretation

A weakness of the approach concerns the data separation in the first layer. Currently a single GP model is trained for all tremor cases, as tremor is reported as a major initial symptom of PD [126]. Hence, the model is not capable of recognizing any other symptom besides tremor, if tremor is present in the data. This means, we miss out on the rarely occurring situations, when patients are co-exhibiting dyskinetic or bradykinetic symptoms. Additionally, we keep track of the stronger symptom among dyskinesia and bradykinesia only and disregard situations, where patients exhibit multiple symptoms in parallel. Besides, in the current approach we assume false positive (FP) and false negative (FN) predictions to be equally undesired. In other medical applications, however, other prediction characteristics could be prioritized, which require the adaptation of the initial GP hyperparameters. To give an example for another FN/FP-ratio: detecting all non-balanced symptom states is three times more important than predicting FPs, what induces a GP model prediction ration of $FN/FP \approx 0.33$. An alternative prediction characteristic is for instance: a symptom should be missed in at most 5% of the estimations. In the present approach, appropriate hyperparamters for such prediction characteristics are found by heuristics.

The results from the previous section, however, show that our multi-layer approach is capable of estimating PD movement symptoms with high precision during daily living activities. We achieve quite precise symptom severity predictions during the activities *sitting, walking, standing, laying* and *other*, indicating that our approach is robust to arbitrary voluntary motion. In comparison to the state of the art in autonomous symptom recognition for PD, we cover a more complete symptom range than most related work by considering tremor, bradykinesia and dyskinesia. Further, we require only one inertial sensor and do not depend on specific tasks and motion patterns for the ambient symptom recognition. Hence, we significantly decrease the inconvenience to the patient during symptom detection and severity estimation.

5.6 Discussion

To conclude this chapter, we briefly recapitulate the characteristics of the developed motor symptom estimation in PD. We introduced an approach to monitor the human motor behavior, and to autonomously detect motor malfunctioning, relying on sensor data from a wearable inertial module.

The essential contribution is the development of a motor symptom detection and severity estimation method, that accurately recognizes symptom severity in the presence of arbitrary and unknown voluntary motion. Hence, the proposed approach is able to determine the patient's symptoms with high precision during daily living, without restrictions on the patient's activities. A key property of the approach is the dynamical system formulation of the fault detection unit (2.16), which is employed to incorporate the temporal evolution of the PD symptoms.

The external fault detection is a framework for estimating the human internal, unknown motor state. We achieve this using a multi-layer GP model given the available motion measurements and a training data set of expert labels of motor symptoms including the synchronized executed motor behavior. We analyze the symptom characteristics to propose expressive inertial data feature and to motivate the multi-layer GP structure. The diversity of motion features based on wavelet decomposition, enables our approach to achieve accurate estimation results during unstructured patient activities. Moreover, as we rely on a commercially available low-cost wearable sensor, handling of the tracking device is user friendly, the device is robust and maintenance is minimal.

We achieve outstanding results in estimating PD patient motor symptoms: In an experimental evaluation, we use a PD patient data set (30 participants) to analyze the recognition accuracy of dyskinesia, bradykinesia with tremor and bradykinesia without tremor. Moreover, we estimate the symptom severity levels in terms of the MDS-UPDRS standard PD rating scale (0-4 per symptom). The testing is performed during unstructured daily living and provides reliable results (accuracy ca. $50 - 90\%$ and ± 1 accuracy ca. 90%) for all (unknown) patient activities; the categories in which the activities were subdivided comprise *walking, sitting, standing, lying* and *other*. Numerous approaches are already available for activity monitoring using motion data from smartwatches. So far we do not make use of activity recognition in our method. In future work, however, we aim to include activity recognition during non-symptom phases and then providing improved symptom detection and severity estimation by filtering out voluntary motions such as cyclical arm swinging during walking, for instance.

6

Conclusion and Future Directions

In this thesis we develop a plant model for the human sensorimotor system. The plant model includes three subsystems: (1) the posterior parietal cortex, where the human decision process takes place, (2) the physical motor execution, where the desired motion trajectory is transformed into motor commands, which traverse the spinal cord to the executing body part and result in muscle activation and coordination, and (3) the motor system, which influences the human motor state. In the following we summarize our achievements and indicate future research directions.

Summary of Contributions

We address all challenges that arise from modeling human motor behavior: We develop novel approximations of the human sensorimotor plant model based on data-driven modeling, since parametric models are proven unsuitable for the complex dynamics of human motor behavior. Further, we generalize the GP model to allow for rigid motion input data, since human executed movements naturally evolve on the special Euclidean group $SE(3)$. And finally, we constantly account for the uncertainty in the human sensorimotor system through incorporating probabilistic approach in all formal methodology.

Motor Execution Approximation

In Chapter 3 we introduce a GP generalization to allow for model input on $SE(3)$. Therefore, we present mean functions and fundamental kernel functions for input data in the space of rigid motions that are based on either the dot function or a metric in the special Euclidean group. Further, we prove those characteristic GP model parts to be valid. To represent the space of the rigid motions, we investigate the most prominent parametrizations and evaluate the GP generalization characteristics depending on the chosen representations, since the GP properties depend on the topology of the parametrization.

We evaluate the generalized GP in simulations and on real word data in applications of human-robot cooperation. The results show the GP generalization to be suitable for approximating the closed loop dynamics of human motor executions, due to their property to generate smooth motion trajectories and to incorporate prediction confidence in the model. From literature, we know that human motor execution is corrupted by various types of sensorimotor uncertainty, which are considered in generalized GP modeling since the predicted output is provided in form of a Gaussian distribution. A method for integrating the probability distributions in velocity space to the space of rigid motions $SE(3)$ allows for a completion of the dynamics approximation of human motor execution.

Decision Process Approximation

The subsystem modeling the human decision making (taking place in the posterior parietal cortex), consisting of a parallel process of potential movement planning and motion weighting, is approximated in Chapter 4. We introduce model approximations for all three units involved in the decision making unit, however, including some adaptations of the subsystem plant model: Since we do not have access to the human internal potential movement plans, we approximate them by executed human movement including sensorimotor noise. We differentiate those examples of human motor behavior using a probabilistic clustering approach based on generalized GMMs, that allow for data inputs on $SE(3)$. For approximating the movement weighting, we introduce a probabilistic classification method, that allows for online trajectory bundle (representing the potential desired trajectories) leveraging. To develop an approximation of the decision making unit in the human subsystem modeling the decision process, we present the online generalized GP model including the sample leveraging. This algorithm enhances the generalized GP to a recursive algorithm that incorporates the sample importance in the kernel function.

The human decision process model is inspired by neurophysiological findings that suggest the decision process dynamics to consist of parallel processes which jointly generate the human internal desired trajectory. The presented approximations certainly represent only one approach out of the numerous possible approximations. In our experimental evaluation, however, the approach performed convincingly. The approach for generating the desired trajectory is capable of predicting a completely unseen rigid motion trajectory online using a generalized GP model. So far, however, we have not evaluated sample leveraging experimentally in the generalized online GP. Further, the performance of the probabilistic classification method is evaluated in comparison to the human. We demonstrate the online algorithm, which relies on the point wise class-conditional likelihood, to significantly outperform the human classification abilities.

Internal Motor State Estimation

In Chapter 5 we provide a monitoring and fault detection system for estimating the human internal motor state. An external unit tracks human executed motor behavior and extracts meaningful features from the captured inertial signals, based on time-frequency transformations, specifically wavelet transformations. Then, a multi-layer GP model estimates the PD patient's motor symptoms employing a dynamical system structure from the previous motor state estimate and the motor behavior features, captured and calculated in the monitoring unit. The multi-layer GP detects the presence of the most salient motor symptoms in PD tremor, dyskinesia and bradykinesia, and estimates the severity of the most prominent of these motor symptoms.

In the development of the motor symptom estimator, we focus on patients with PD and therefore, solely regarded motor dysfunctions that appear for Parkinson's patients. The experimental evaluation on real patients' data of those motor symptoms show clearly superior performance compared to the state of the art in automated symptom recognition in PD: We not only detect the presence of diverse motor symptoms, but also estimated the severity of those symptoms. Additionally, we allow the patient to perform unconstrained activities of daily living. Hence, the fault detection achieved reliable estimation results of motor symptoms besides the presence of unknown intended motions in the captured data.

Outlook

In this thesis we aim towards holistically modeling the human sensorimotor system. We introduce for each of the dynamical subsystems in the human plant model a novel approximation based on GP models, formal results on the methodology and thorough experimental evaluations and illustrations of the presented approximations in the areas of human-robot cooperation and health-care, respectively.

Although many issues in modeling the human sensorimotor system have been addressed in this thesis, a number of research problems remain to be solved in future work.

Model Generalizations to Manifolds

In future work, we aim towards introducing the generalization of GP modeling for the rigid motion representation by homogeneous matrices. The way was paved already in this thesis for the generalization to allow for parametrizing GPs in matrix form. It only remains open to chose a suitable distance measure on homogeneous matrices for to complete the approach. Another interesting line of research is the GP generalization to other manifolds than $SE(3)$. So far, no kernel function has been introduced for modeling correlation among data points on a less restrictive manifold topology. Additionally, we seek to employ the generalization fundamentals to other machine learning techniques such as neural networks for instance.

Further Analysis of Human Decision Making

The evaluation of reasons for the poor classification results of humans in the goal configuration prediction experiment, remains open. The gap between the human behavior and the algorithm might as well indicate that our model of the human decision process does not fully capture the true process in the human brain, which results in a desired trajectory. Further, we aim to finalize the human-robot cooperation experiment concerning the putting on of a shoe to a human foot. Since, the demographic change in most western countries tends to a society, which grows older, we will require more automated elderly assistance in the future. Therefore, we are looking forward to further investigations on the topic of robots as dressing assistants.

Motor State Estimation for general Motor Abnormalities

We believe that the ambient symptom and motor state recognition of patients with motor abnormalities has great potential. This research field is relatively new to the community and especially when it comes to applying advanced machine learning techniques for nonlinear modeling, only a few approaches have been applied so far. Many diseases including motor dysfunctions besides PD, exhibit characteristic symptoms. Wearable inertial sensors allow for capturing those features non-invasively. Then, advanced machine learning techniques such as deep neural nets or the presented multi-layer GP can be trained to identify symptoms including their severities with high accuracy. However, it remains to be demonstrated in future work, what accuracy of the motor symptom estimation can be achieved in other fault detection systems.

List of Figures

2.1 Control scheme of relevant aspects of the human sensorimotor system and motor output. Subsystem 2 concerns the decision making for a spatial movement action and takes place in the posterior parietal cortex of the human brain. Subsystem 1 concerns the subsequent movement execution of the planned action. The transformation from the desired trajectory to the actual end-point movement can have a delay of up to 40 ms. Under certain circumstances the motor control can be impaired, namely, when the internal state of the motor system is dysfunctional. Subsystem 3 models the state of the motor system. The last remaining subsystem $\hat{3}$ is an external system monitoring the human motor behavior to estimate the human internal motor system state. 10

2.2 Scheme of the noisy nonlinear motion dynamics mapping from the space $\mathfrak{g} \in SE(3)$ to the corresponding velocity space $T_{\mathfrak{g}}SE(3)$. 13

2.3 Scheme of the decision process consisting of parallel motion prediction on $SE(3)$ and estimation of the belonging class. 16

2.4 Scheme of the symptom level estimation. The internal motor state ξ_κ corrupts the intended motion in case it is unbalanced. Then, the resulting executed movement $\mathfrak{g}_\kappa$ includes motor dysfunctions. The monitoring system not only captures the signal, but also generates features $\mathbf{z}_\kappa$ suitable to characterize the motion. The fault detection estimates the level of unintended motion in the captured signal and hence provides an estimate $\hat{\xi}_\kappa$ for the symptom level. . . 18

3.1 The distance measures consisting of quaternion norm, arc length and projection to the tangent space are depicted by a red dashed, green solid and blue dashed-dotted line, respectively. 32

3.2 The Gaussian distribution in the tangent plane to the north pole is projected to the sphere S_2 via central projection. Hence, each point in the plane is projected to both opposite intersections of the sphere with a straight line through the center of the sphere and the point in the tangent plane. 39

3.3 The Gaussian distribution depicted by the dashed blue line is projected via the sphere from the tangent space at $\mathbf{q}_{\text{rot}} = (0,\, 1)$ into the tangent space at $\mathbf{q}'_{\text{rot}} = (0.3,\, 0.95)$. After projection, the red curve is approximated using the unscented transform by a Gaussian, visualized by the green dashed-dotted line. 43

3.4 Numerical performance analysis in predicting dynamics f_M. The bars depict the average RMSEs in the rotational velocity prediction, where solid red colors show the $\mathrm{GP}_{\mathbb{H}_D}$ and striped blue the classic GP. The standard deviation among the ten different dynamics is shown by the error bars. 48

3.5 A human participant repetitively draws large “Z” letters on a flip chart, while naturally tracking the pen with the eye. The human gaze motion is captured in workspace coordinates using a combination of eye tracking and marker based head motion tracking. 49

3.6 The prediction accuracy of the human gaze motion is illustrated. Blue solid lines show the mean prediction, red dashed lines the ground truth and gray shaded areas the prediction uncertainty. The columns correspond to the tangent bundle respectively rotational velocity dimensions, the rows to the different conditions. All columns are synchronized over time. In row one the GP model is based on the arc distance, in row two on the quaternion norm, in row three on the distance projected to the tangent space and in row four on the Euclidean distance in combination with the Euler angle representation. Column wise a second trajectory is highlighted by yellow background color. . 50

3.7 Human-human shoe providing experiment. One human puts on a shoe to another as preliminary study for elderly assistance in service robotics. The blue lines depict the translation part of the providers’ 6D motion trajectories. 51

3.8 Visualization of the training data (colored dotted), the newly generated motion trajectory by GP prediction (dashed black), the mean of human demonstrations (red solid) and the standard GP prediction (blue dash-dotted). . . 52

3.9 Runtime and prediction accuracy comparison. Per GP variant the box height shows the average required time and the average root-mean-square error (RMSE), respectively. The gray error bar depict the standard deviation over the 1000 experiment repetitions. 54

4.1 One human puts on a shoe to another in a human-human cooperation study to prepare intuitive elderly assistance in service robotics. A motion tracking system captures the provider’s adaptation of the shoe to the receiver’s leg motion and a 6-DoF force and torque sensor captures the contact forces during the slipping over motion. 73

4.2 Successful robotic lower limb assistance. A robot approaches the human shoe receiver, positions the shoe relative to the estimated receiver’s goal foot pose and successfully slips the red shoe over the foot. 73

4.3 A natural 6D human handover is visualized. The object handover starts from an initial hand configuration, performs the object grasp and terminates at the final configuration after the object is handed over. The handover motion in this trial incorporates a half turn of the object. 74

4.4 Estimation results of the generalized online GP algorithm. The vertical bar indicates the current time stamp. In the past, left of the bar, the Gaussian posterior is available, whereas on its right, in the future, the motion is predicted. The first row depicts the translational velocities, the second row the tangent space velocities and the third row shows the imaginary components of a quaternion, which is obtained by projecting the velocities of the row above to the sphere. 76

4.5 Bayesian information criterion (BIC) value for GMM with cluster cardinality form 1 to 10. On the left, the average and standard deviation of the BIC values over 70 experiment repetitions are depicted; on the right the lowest BIC values obtained throughout all 70 experiments are shown for each cluster size. In both settings, the optimal number of GMM clusters $\hat{\varrho}$, depicted in red, is 5. 78

4.6 Histogram of the optimal obtained BIC values over 70 experiment repetitions. Clearly, the GMMs with 5 clusters generated the lowest BIC value in most cases (25 out of 70). 78

4.7 Histogram over the final foot clusters performed by each of the 5 subjects. The distribution over the goal configurations shows the diversity in personal preference of the individuals but no correlation between the clusters and personal attributes of the subjects. 79

4.8 Generalized GMM clusters for which we obtained the globally optimal BIC value 3286.56. The receiver's foot goal position is illustrated by circles, colored according to the most probable cluster membership. They are presented in human chest frame, and thus, depend on the receiver's sitting posture. Even though only the data positions are visualized for clarity reasons, the generalized GMM clustering was performed on the full rigid motions $\mathfrak{g}_j$, $j = 1, \ldots, \hat{\varrho}$. The ellipsoids represent the fitted cluster components. Each projected Gaussian is centered at one of the coordinate frames *right, left, bent, cross low, cross high* and spans the corresponding ellipsoid with it's 2σ-uncertainty region. 80

4.9 Trajectories of the shoe in 3D space. The color gradient of the trajectories depict the evolution of the movement from the start (dark blue) to the goal configurations, when the foot is successfully inserted in the shoe (yellow). The same set of trajectories, which is represented in a global frame on the left, is represented in the goal foot frame on the right. The goal foot frame is the configuration, in which the shoe insertion in that specific cluster has taken place. 81

4.10 Norm of the translational velocity trajectories of the shoe. On the left side, the mean and the standard deviation of the velocities per condition are visualized over time, and on the right side, the variability in the execution velocity is depicted over the distance to the foot. 82

4.11 Goal configurations of the 5 clusters *left, right, cross high, cross low* and *bent.* The pictures are shown to the participants as representatives of the possible clusters. 84

4.12 The first video frames after the online classification had correctly classified the synchronized movement trajectories into the conditions *cross high* respectively *left*. The algorithm is based on the foot motion only and does not take the knee movement into account. 86

5.1 Wrist-worn inertial sensor Microsoft Band 2. The band is placed medially at the patient's wrist as displayed to assure the reference frames of all measurements to match. 93

5.2 Distribution of symptom severities and activities of the participants. The label distributions were calculated from the collected dataset for each participant and then averaged. The arrows in the symptom chart (left) indicate increasing severity of dyskinesia (DK) and bradykinesia (BK) scores in 4 levels each. The tremor class (TM) as well has four severity levels, but is visualized here as a single section (purple). 94

5.3 Power spectral density (PSD) of the accelerometer data. The PSD is visualized for bradykinesia (BK), tremor (TM), dyskinesia (DK) and the balanced condition (none). It shows generally high densities during walking, where the dotted line depicts the scale limit of the other three plots, and very low PSD values during laying. 95

5.4 Average PSD of the non-tremor accelerometer data per symptom severity. The mean (visualized by bar height) of each PSD per symptom severity level with its standard deviation (gray error bar) is sorted from balanced to severe. In both figures the yellow bar depicts the average PSD level in the balanced condition. On the left side the symptom levels of bradykineasia (without tremor) are visualized in blue shades, on the right side the levels of dyskinesia in red shades. 95

5.5 Autocorrelation of symptom labels. Autocorrelation coefficients of the bradykinesia, dyskinesia and tremor fluctuations indicate a very slow transition of dys- and bradykinesia severity whereas the transition is comparably faster for tremor severity. The data were averaged across the (available) patients, and the error bars indicate one standard deviation. 96

5.6 Inertial data and expert ratings processing. The sensor data is pre-processed by applying a two-directional Butterworth filter and wavelet decomposition, before clinically relevant PD features are extracted. The result is then synchronized with the activity and symptom severity labels of the expert rater. 96

5.7 Wavelet transformation of the accelerometer data. The discrete wavelet decomposition is performed on the vector norm of 3D accelerometer (and gyroscope) measurements to calculate features (e.g. standard deviation, norm and root mean square) that describe characteristics of the 60 seconds time window. Only the third level of the accelerometer signal decomposition is depicted. The green shaded area indicates presence of the tremor label. . . . 98

5.8 Scheme of multi-layer algorithm. The feature vector $\mathbf{z}_\kappa$ and the state estimate $\xi_{\kappa-1}$ are the inputs to the multi-layer approach. After testing for tremor in the first model layer ($y_{\mathsf{TM}} < 0.5$ or $y_{\mathsf{TM}} \geq 0.5$), the remaining non-tremor data is tested in the second layer for dyskinesia and bradykinesia severity. Each of the GPs provides an estimate of the symptom severity in the continuous space $\mathbb{R}$, where, however, values outside the interval $[-0.5, 4.49]$ are extremely rare. In the third layer of the hierachical approach we decide among the three remaining motor symptom conditions balanced, bradykinesia and dyskinesia depending on the estimated symptom severities. If both predictions obtained from the dyskinesia GP and the bradykinesia GP, respectively, are below a certain threshold ($\tilde{a} = 0.5$), the balanced condition is nominated. If any of the predicted values exceeds the threshold, the symptom class of the larger value is appointed. To determine one of the categorical symptom severity levels $1 - 4$, the GP prediction is rounded to an integer value. 100

5.9 Distribution of symptom severities according to the different rating scales. According to the independent scales all data where a specific symptom is not apparent is assigned to the balanced class. To reduce the non-uniformity of the symptom distribution during the GP model training, the dyskinesia GP, the bradykinesia GP and the tremor GP are trained on the data subsets illustrated in the bottom row left, middle and right, respectively. For all three GPs the model testing, however, is performed on the full data sets (top row). 103

List of Tables

4.1 Overview over the classification results broken down according to the conditions. 84
4.2 Overview over all 1100 repetitions broken down according to the conditions. 85

5.1 Training accuracy of the symptom severity for each activity. 104
5.2 Individual layer testing accuracy of the symptom severity for each activity. . 106
5.3 Total testing accuracy of the symptom severity for each activity. 106
5.4 Individual layer testing accuracy of the symptom severity for each activity without dynamics. 107

Bibliography

[1] N. Sebanz and G. Knoblich, “Prediction in Joint Action: What, When, and Where,” *Topics in Cognitive Science*, vol. 1, no. 2, pp. 353–367, apr 2009. [Online]. Available: http://doi.wiley.com/10.1111/j.1756-8765.2009.01024.x

[2] D. W. Franklin and D. M. Wolpert, “Computational Mechanisms of Sensorimotor Control,” *Neuron*, vol. 72, no. 3, pp. 425–442, nov 2011. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0896627311008919

[3] J. Gordon, M. F. Ghilardi, and C. Ghez, “Accuracy of planar reaching movements,” *Experimental Brain Research*, vol. 99, no. 1, pp. 97–111, may 1994. [Online]. Available: http://link.springer.com/10.1007/BF00241415

[4] M. M. Churchland, A. Afshar, and K. V. Shenoy, “A Central Source of Movement Variability,” *Neuron*, vol. 52, no. 6, pp. 1085–1096, dec 2006. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0896627306008713

[5] R. J. van Beers, P. Haggard, and D. M. Wolpert, “The Role of Execution Noise in Movement Variability,” *Journal of Neurophysiology*, vol. 91, no. 2, pp. 1050–1063, feb 2004. [Online]. Available: http://www.physiology.org/doi/10.1152/jn.00652.2003

[6] A. A. Faisal, L. P. J. Selen, and D. M. Wolpert, “Noise in the nervous system,” *Nature Reviews Neuroscience*, vol. 9, no. 4, pp. 292–303, apr 2008. [Online]. Available: http://www.nature.com/articles/nrn2258

[7] R. A. Scheidt, J. B. Dingwell, F. A. Mussa-ivaldi, A. Robert, J. B. Dingwell, and A. Ferdinando, “Learning to Move Amid Uncertainty,” *Journal of Neurophysiology*, no. 86, pp. 971–985, 2001. [Online]. Available: http://jn.physiology.org/content/86/2/971.short

[8] C. D. Takahashi, R. a. Scheidt, and D. J. Reinkensmeyer, “Impedance Control and Internal Model Formation When Reaching in a Randomly Varying Dynamical Environment,” *Journal of Neurophysiology*, vol. 86, no. 2, pp. 1047–1051, aug 2001. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/11495973 http://www.physiology.org/doi/10.1152/jn.2001.86.2.1047

[9] W. M. Murray, S. L. Delp, and T. S. Buchanan, “Variation of muscle moment arms with elbow and forearm position,” *Journal of Biomechanics*, vol. 28, no. 5, pp. 513–525, may 1995. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/002192909400114J

[10] J. M. Hollerbach and T. Flash, “Dynamic interactions between limb segments during planar arm movement,” *Biological Cybernetics*, vol. 44, no. 1, pp. 67–77, may 1982. [Online]. Available: http://link.springer.com/10.1007/BF00353957

[11] J. R. Medina, M. Lawitzky, A. Mörtl, D. Lee, and S. Hirche, “An Experience-driven Robotic Assistant Acquiring Human Knowledge to Improve Haptic Cooperation,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, sep 2011, pp. 2416–2422.

[12] K. P. Körding and D. M. Wolpert, “Bayesian integration in sensorimotor learning,” *Nature*, vol. 427, no. 6971, pp. 244–247, jan 2004. [Online]. Available: http://www.nature.com/articles/nature02169

[13] ——, “Bayesian decision theory in sensorimotor control,” *Trends in Cognitive Sciences*, vol. 10, no. 7, pp. 319–326, jul 2006. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S1364661306001276

[14] M. Jazayeri and M. N. Shadlen, “Temporal context calibrates interval timing,” *Nature Neuroscience*, vol. 13, no. 8, pp. 1020–1026, aug 2010. [Online]. Available: http://www.nature.com/articles/nn.2590

[15] M. Haruno and D. M. Wolpert, “Optimal Control of Redundant Muscles in Step-Tracking Wrist Movements,” *Journal of Neurophysiology*, vol. 94, no. 6, pp. 4244–4255, dec 2005. [Online]. Available: http://www.physiology.org/doi/10.1152/jn.00404.2005

[16] D. A. Braun, A. Aertsen, D. M. Wolpert, and C. Mehring, “Learning Optimal Adaptation Strategies in Unpredictable Motor Tasks,” *Journal of Neuroscience*, vol. 29, no. 20, pp. 6472–6478, may 2009. [Online]. Available: http://www.jneurosci.org/cgi/doi/10.1523/JNEUROSCI.3075-08.2009

[17] B. Corteville, E. Aertbelien, H. Bruyninckx, J. De Schutter, and H. Van Brussel, “Human-inspired robot assistant for fast point-to-point movements,” in *Proceedings 2007 IEEE International Conference on Robotics and Automation*. IEEE, apr 2007, pp. 3639–3644. [Online]. Available: http://ieeexplore.ieee.org/document/4209654/

[18] N. Jarrasse, J. Paik, V. Pasqui, and G. Morel, “How can human motion prediction increase transparency?” in *IEEE International Conference on Robotics and Automation*, may 2008, pp. 2134–2139.

[19] C. G. Atkeson and J. M. Hollerbach, “Kinematic features of unrestrained vertical arm movements.” *The Journal of neuroscience : the official journal of the Society for Neuroscience*, vol. 5, no. 9, pp. 2318–30, sep 1985. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/4031998

[20] D. W. Franklin, U. So, M. Kawato, and T. E. Milner, “Impedance Control Balances Stability With Metabolically Costly Muscle Activation,” *Journal of Neurophysiology*, vol. 92, no. 5, pp. 3097–3105, nov 2004. [Online]. Available: http://jn.physiology.org/cgi/doi/10.1152/jn.00364.2004 http://www.physiology.org/doi/10.1152/jn.00364.2004

[21] D. R. Lametti, G. Houle, and D. J. Ostry, “Control of Movement Variability and the Regulation of Limb Impedance,” *Journal of Neurophysiology*, vol. 98, no. 6, pp. 3516–3524, dec 2007. [Online]. Available: http://www.physiology.org/doi/10.1152/jn.00970.2007

[22] G. Ariff, O. Donchin, T. Nanayakkara, and R. Shadmehr, “A real-time state predictor in motor control: study of saccadic eye movements during unseen reaching movements.” *The Journal of neuroscience : the official journal of the Society for Neuroscience*, vol. 22, no. 17, pp. 7721–9, sep 2002. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/12196595

[23] B. Mehta and S. Schaal, “Forward Models in Visuomotor Control,” *Journal of Neurophysiology*, vol. 88, no. 2, pp. 942–953, aug 2002. [Online]. Available: http://www.physiology.org/doi/10.1152/jn.2002.88.2.942

[24] D. Wolpert, Z. Ghahramani, and M. Jordan, “An internal model for sensorimotor integration,” *Science*, vol. 269, no. 5232, pp. 1880–1882, sep 1995. [Online]. Available: http://www.sciencemag.org/cgi/doi/10.1126/science.7569931

[25] M. Desmurget, H. Gréa, J. S. Grethe, C. Prablanc, G. E. Alexander, and S. T. Grafton, “Functional anatomy of nonvisual feedback loops during reaching: a positron emission tomography study.” *The Journal of neuroscience : the official journal of the Society for Neuroscience*, vol. 21, no. 8, pp. 2919–28, apr 2001. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/11306644

[26] S. Miossec and A. Kheddar, “Human motion in cooperative tasks: Moving object case study,” in *2008 IEEE International Conference on Robotics and Biomimetics*, no. i. IEEE, feb 2009, pp. 1509–1514. [Online]. Available: http://ieeexplore.ieee.org/document/4913224/

[27] E. Todorov and M. I. Jordan, “Optimal feedback control as a theory of motor coordination.” *Nature neuroscience*, vol. 5, no. 11, pp. 1226–1235, 2002.

[28] D. W. Franklin, E. Burdet, K. Peng Tee, R. Osu, C.-M. Chew, T. E. Milner, and M. Kawato, “CNS Learns Stable, Accurate, and Efficient Movements Using a Simple Algorithm,” *Journal of Neuroscience*, vol. 28, no. 44, pp. 11 165–11 173, oct 2008. [Online]. Available: http://www.jneurosci.org/cgi/doi/10.1523/JNEUROSCI.3099-08.2008

[29] M. Kawato, “Internal models for motor control and trajectory planning,” *Current Opinion in Neurobiology*, vol. 9, no. 6, pp. 718–727, dec 1999. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0959438899000288

[30] R. L. Klatzky, T. G. Fikes, and J. W. Pellegrino, “Planning for hand shape and arm transport when reaching for objects.” *Acta psychologica*, vol. 88, no. 3, pp. 209–232, 1995.

[31] S. Endo, A. M. Wing, and R. M. Bracewell, “Haptic and visual influences on grasp point selection.” *Journal of motor behavior*, vol. 43, no. 6, pp. 427–431, jan 2011. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/22004190

[32] Y. Paulignan, V. G. Frak, I. Toni, and M. Jeannerod, “Influence of object position and size on human prehension movements.” *Experimental brain research. Experimentelle Hirnforschung. Experimentation cerebrale*, vol. 114, no. 2, pp. 226–234, 1997.

[33] M. Desmurget, C. Prablanc, M. Arzi, Y. Rossetti, Y. Paulignan, and C. Urquizar, “Integrated control of hand transport and orientation during prehension movements.” *Experimental brain research*, vol. 110, no. 2, pp. 265–278, jul 1996. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/8836690

[34] D. A. Rosenbaum, J. Jorgensen, and M. J. Jorgensen, “Planning macroscopic aspects of manual control,” *Human Movement Science*, vol. 11, no. 1-2, pp. 61–69, 1992.

[35] B. L. Smith, B. M. Williams, and R. K. Oswald, “Comparison of Parametric and Nonparametric Models for Traffic Flow Forecasting,” *Transportation Research Part C: Emerging Technologies*, vol. 10, no. 4, pp. 303–321, 2002.

[36] J. Wang, D. Fleet, and A. Hertzmann, “Gaussian Process Dynamical Models for Human Motion,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 2, pp. 283–298, feb 2008. [Online]. Available: http://ieeexplore.ieee.org/document/4359316/

[37] S. M. Khansari-Zadeh and A. Billard, “Learning Stable Nonlinear Dynamical Systems With Gaussian Mixture Models,” *IEEE Transactions on Robotics*, vol. 27, no. 5, pp. 943–957, oct 2011.

[38] D. Kulić, W. Takano, and Y. Nakamura, “Incremental learning of full body motions via adaptive factorial hidden markov models,” in *7th International Conference on Epigenetic Robotics*, 2007.

[39] Y. Bengio, P. Simard, and P. Frasconi, “Learning long-term dependencies with gradient descent is difficult,” *IEEE Transactions on Neural Networks*, vol. 5, no. 2, pp. 157–166, mar 1994. [Online]. Available: http://ieeexplore.ieee.org/document/279181/

[40] S. Hochreiter, “The Vanishing Gradient Problem During Learning Recurrent Neural Nets and Problem Solutions,” *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 06, no. 02, pp. 107–116, apr 1998. [Online]. Available: http://www.worldscientific.com/doi/abs/10.1142/S0218488598000094

[41] I. Goodfellow, Y. Bengio, and A. Courville, “Deep Learning,” *MIT Press*, 2016. [Online]. Available: http://www.deeplearningbook.org/

[42] S. Genc, “Parametric system identification using deep convolutional neural networks,” in *2017 International Joint Conference on Neural Networks (IJCNN)*. IEEE, may 2017, pp. 2112–2119. [Online]. Available: http://ieeexplore.ieee.org/document/7966110/

[43] C. E. Rasmussen and C. K. I. Williams, *Gaussian processes for machine learning.* MIT Press, 2006, vol. 14, no. 2. [Online]. Available: http://www.gaussianprocess.org/gpml/chapters/RW.pdf

[44] G. S. Aoude, B. D. Luders, J. M. Joseph, N. Roy, and J. P. How, “Probabilistically safe motion planning to avoid dynamic obstacles with uncertain motion patterns,” *Autonomous Robots*, vol. 35, no. 1, pp. 51–76, jul 2013. [Online]. Available: http://link.springer.com/10.1007/s10514-013-9334-3

[45] M. Lang and W. Feiten, “MPG - Fast Forward Reasoning on 6 DOF Pose Uncertainty,” in *Robotics; Proceedings of ROBOTIK 2012; 7th German Conference on*, may 2012, pp. 1–6.

[46] M. Lang, O. Dunkley, and S. Hirche, “Gaussian process kernels for rotations and 6D rigid body motions,” in *IEEE ICRA 2014 International Conference on Robotics and Automation*. IEEE, may 2014, pp. 5165–5170. [Online]. Available: http://ieeexplore.ieee.org/document/6907617/

[47] M. Lang, M. Kleinsteuber, O. Dunkley, and S. Hirche, “Gaussian process dynamical models over dual quaternions,” in *2015 European Control Conference (ECC)*. Linz, Austria: IEEE, jul 2015, pp. 2847–2852. [Online]. Available: http://ieeexplore.ieee.org/document/7330969/

[48] M. Lang, M. Kleinsteuber, and S. Hirche, “Gaussian process for 6-DoF rigid motions,” *Autonomous Robots*, pp. 1–17, nov 2017. [Online]. Available: http://link.springer.com/10.1007/s10514-017-9683-4

[49] M. Lang and S. Hirche, “Computationally Efficient Rigid-Body Gaussian Process for Motion Dynamics,” *IEEE Robotics and Automation Letters*, vol. 2, no. 3, pp. 1601–1608, jul 2017. [Online]. Available: http://ieeexplore.ieee.org/document/7869309/

[50] W. Feiten, M. Lang, and S. Hirche, “Rigid Motion Estimation using Mixtures of Projected Gaussians,” in *16th International Conference on Information Fusion (FUSION)*, Istanbul, Turkey, jul 2013, pp. 1465–1472. [Online]. Available: http://www.r3-cop.eu/wp-content/uploads/2013/07/Rigid-Motion-Estimation-using-Mixtures-of-Projected-Gaussians.pdf

[51] M. Lang, F. M. Pfister, J. Fröhner, K. Abedinpour, D. Pichler, U. Fietzek, T. T. Um, D. Kulić, S. Endo, and S. Hirche, “A Multi-layer Gaussian Process for Motor Symptom Estimation in People with Parkinson’s Disease,” *IEEE Transactions on Biomedical Engineering*, feb 2019.

[52] S. Kakei, D. S. Hoffman, and P. L. Strick, “Sensorimotor transformations in cortical motor areas,” *Neuroscience Research*, vol. 46, no. 1, pp. 1–10, may 2003. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0168010203000312

[53] R. A. Andersen and H. Cui, “Intention, Action Planning, and Decision Making in Parietal-Frontal Circuits,” *Neuron*, vol. 63, no. 5, pp. 568–583, sep 2009. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0896627309006394

[54] A. H. Fagg and M. A. Arbib, “Modeling parietal–premotor interactions in primate control of grasping,” *Neural Networks*, vol. 11, no. 7-8, pp. 1277–1303, oct 1998. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0893608098000471

[55] P. Cisek, “Cortical mechanisms of action selection: the affordance competition hypothesis,” in *Modelling Natural Action Selection*, A. Seth, T. Prescott, and J. Bryson, Eds. Cambridge: Cambridge University Press, 2011, pp. 208–238. [Online]. Available: https://www.cambridge.org/core/product/identifier/CBO9780511731525A022

[56] M. N. Shadlen and W. T. Newsome, “Neural basis of a perceptual decision in the parietal cortex (area LIP) of the rhesus monkey.” *Journal of neurophysiology*, vol. 86, no. 4, pp. 1916–36, oct 2001. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/11600651

[57] G. H. Mulliken, S. Musallam, and R. A. Andersen, “Forward estimation of movement state in posterior parietal cortex,” *Proceedings of the National Academy of Sciences*, vol. 105, no. 24, pp. 8170–8177, jun 2008. [Online]. Available: http://www.pnas.org/cgi/doi/10.1073/pnas.0802602105

[58] K. Friston, “What Is Optimal about Motor Control?” *Neuron*, vol. 72, no. 3, pp. 488–498, nov 2011. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0896627311009305

[59] G. Orbán and D. M. Wolpert, “Representations of Uncertainty in Sensorimotor Control,” *Current Opinion in Neurobiology*, vol. 21, no. 4, pp. 629–635, aug 2011. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0959438811000948

[60] J. Ko and D. Fox, “GP-Bayes Filters: Bayesian filtering using Gaussian process prediction and observation models,” in *IROS 2008. IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2008.

[61] R. Subbarao, “Robust Statistics over Riemannian Manifolds for Computer Vision,” Ph.D. dissertation, Rutgers University; Graduate School - New Brunswick, 2008. [Online]. Available: http://dx.doi.org/doi:10.7282/T3736R88

[62] C. G. Goetz, B. C. Tilley, S. R. Shaftman, G. T. Stebbins, S. Fahn, P. Martinez-Martin, W. Poewe, C. Sampaio, M. B. Stern, R. Dodel, B. Dubois, R. Holloway, J. Jankovic, J. Kulisevsky, A. E. Lang, A. Lees, S. Leurgans, P. A. LeWitt, D. Nyenhuis, C. W. Olanow, O. Rascol, A. Schrag, J. A. Teresi, J. J. van Hilten, and N. LaPelle, “Movement Disorder Society-sponsored revision of the Unified Parkinson’s Disease Rating Scale (MDS-UPDRS): Scale presentation and clinimetric testing results,” *Movement Disorders*, vol. 23, no. 15, pp. 2129–2170, nov 2008. [Online]. Available: http://doi.wiley.com/10.1002/mds.21198 http://doi.wiley.com/10.1002/mds.22340

[63] C. K. I. Williams and C. E. Rasmussen, “Gaussian Processes for Regression,” *Advances in Neural Information Processing Systems 8*, vol. 8, no. August, pp. 514–520, 1996. [Online]. Available: http://eprints.aston.ac.uk/651/

[64] R. Neal, *Bayesian Learning for Neural Networks.* Springer-Verlag, 1996. [Online]. Available: https://www.springer.com/de/book/9780387947242

[65] M. N. Gibbs, “Bayesian Gaussian processes for regression and classification,” *PhD dissertation, University of Cambridge*, 1997. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.147.1130

[66] C. M. Bishop, “Pattern Recognition and Machine Learning,” *Journal of Electronic Imaging*, vol. 16, no. 4, p. 049901, jan 2007. [Online]. Available: http://electronicimaging.spiedigitallibrary.org/article.aspx?doi=10.1117/1.2819119

[67] H. Kang and F. C. Park, “Motion optimization using Gaussian process dynamical models,” *Multibody System Dynamics*, vol. 34, no. 4, pp. 307–325, 2015.

[68] L. Ljung, “System Identification,” in *Wiley Encyclopedia of Electrical and Electronics Engineering.* Hoboken, NJ, USA: John Wiley & Sons, Inc., dec 1999, vol. 19, no. 2, pp. 163–173.

[69] G. Pillonetto and G. De Nicolao, “A new Kernel-based Approach for Linear System Identification,” *Automatica*, vol. 46, no. 1, pp. 81–93, 2010.

[70] K. Fukumizu, A. Gretton, B. Schölkopf, and B. K. Sriperumbudur, “Characteristic Kernels on Groups and Semigroups,” in *Advances in Neural Information Processing Systems 21*, D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, Eds. Curran Associates, Inc., 2009, pp. 473–480.

[71] M. Harandi, M. Salzmann, and F. Porikli, “Bregman Divergences for Infinite Dimensional Covariance Matrices,” in *2014 IEEE Conference on Computer Vision and Pattern Recognition.* IEEE, jun 2014, pp. 1003–1010. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6909528

[72] S. Jayasumana, R. Hartley, M. Salzmann, H. Li, and M. Harandi, “Kernel Methods on the Riemannian Manifold of Symmetric Positive Definite Matrices,” in *2013 IEEE Conference on Computer Vision and Pattern Recognition.* IEEE, jun 2013, pp. 73–80. [Online]. Available: http://ieeexplore.ieee.org/document/6618861/

[73] L. Song, K. Fukumizu, and A. Gretton, “Kernel Embeddings of Conditional Distributions: A Unified Kernel Framework for Nonparametric Inference in Graphical Models,” *IEEE Signal Processing Magazine*, vol. 30, no. 4, pp. 98–111, jul 2013. [Online]. Available: http://ieeexplore.ieee.org/document/6530747/

[74] J. B. Tenenbaum, V. de Silva, and J. C. Langford, “A Global Geometric Framework for Nonlinear Dimensionality Reduction,” *Science*, vol. 290, no. 5500, p. 2319, 2000.

[75] S. T. Roweis and L. K. Saul, “Nonlinear Dimensionality Reduction by Locally Linear Embedding,” *Science*, vol. 290, no. 5500, pp. 2323–2326, 2000.

[76] M. Belkin and P. Niyogi, “Laplacian Eigenmaps for Dimensionality Reduction and Data Representation,” *Neural Computation*, vol. 15, no. 6, pp. 1373–1396, jun 2003. [Online]. Available: http://www.mitpressjournals.org/doi/10.1162/089976603321780317

[77] J. Nilsson, F. Sha, and M. I. Jordan, “Regression on Manifolds Using Kernel Dimension Reduction,” in *Proceedings of the 24th International Conference on Machine Learning*, ser. ICML ’07. New York, NY, USA: ACM, 2007, pp. 697–704.

[78] J. Ham, D. D. Lee, S. Mika, and B. Schölkopf, “A Kernel View of the Dimensionality Reduction of Manifolds,” in *Proceedings of the Twenty-first International Conference on Machine Learning*, ser. ICML ’04. New York, NY, USA: ACM, 2004.

[79] R. Urtasun, D. Fleet, A. Geiger, J. Popovic, T. Darrell, and N. Lawrence, “Topologically-Constrained Latent Variable Models,” in *International Conference on Machine Learning (ICML)*, 2008.

[80] E. del Castillo, B. M. Colosimo, and S. D. Tajbakhsh, “Geodesic Gaussian Processes for the Parametric Reconstruction of a Free-Form Surface,” *Technometrics*, vol. 57, no. 1, pp. 87–99, jan 2015. [Online]. Available: http://www.tandfonline.com/doi/abs/10.1080/00401706.2013.879075

[81] R. Calandra, J. Peters, C. E. Rasmussen, and M. P. Deisenroth, “Manifold Gaussian Processes for regression,” in *2016 International Joint Conference on Neural Networks (IJCNN).* IEEE, jul 2016, pp. 3338–3345. [Online]. Available: http://ieeexplore.ieee.org/document/7727626/

[82] F. Thomas, “Approaching Dual Quaternions From Matrix Algebra,” *IEEE Transactions on Robotics*, vol. 30, no. 5, pp. 1037–1048, oct 2014.

[83] E. Ata, Y. Yayli, and Y. Ata Erhan andf Yayli, “Dual Unitary Matrices and Unit Dual Quaternions,” *Differential Geometry - Dynamical Systems*, vol. 10, no. Vol 10, pp. 1–12, 2008. [Online]. Available: http://www.mathem.pub.ro/dgds/v10/D10-AA.pdf

[84] B. Palais and R. Palais, “Euler’s fixed point theorem: The axis of a rotation,” *Journal of Fixed Point Theory and Applications*, vol. 2, no. 2, pp. 215–220, dec 2007. [Online]. Available: http://link.springer.com/10.1007/s11784-007-0042-5

[85] Y. Matsuoka, H. Durrant-Whyte, and J. Neira, *Robotics: Science and Systems VI.* The MIT Press, 2011. [Online]. Available: https://books.google.de/books?id=JQqPKQEACAAJ

[86] D. Q. Huynh, “Metrics for 3D Rotations: Comparison and Analysis,” *Journal of Mathematical Imaging and Vision*, vol. 35, no. 2, pp. 155–164, oct 2009. [Online]. Available: http://link.springer.com/10.1007/s10851-009-0161-2

[87] F. C. Park, “Distance Metrics on the Rigid-Body Motions with Applications to Mechanism Design,” *Journal of Mechanical Design*, vol. 117, no. 1, p. 48, 1995. [Online]. Available: http://mechanicaldesign.asmedigitalcollection.asme.org/article.aspx?articleid=1444402

[88] D. Duvenaud, J. R. Lloyd, R. Grosse, J. B. Tenenbaum, and Z. Ghahramani, “Structure Discovery in Nonparametric Regression through Compositional Kernel Search,” in *International Conference on Machine Learning*, 2013.

[89] D. Zhang, X. Chen, and W. S. Lee, “Text Classification with Kernels on the Multinomial Manifold,” in *International ACM SIGIR Conference on Research and Development in Information Retrieval.* New York, NY, USA: ACM Press, 2005, pp. 266–273.

[90] P. Neumann, “The role of geodesic and stereographic projections for the visualization of directions, rotations, and textures,” *Physica Status Solidi (a)*, vol. 131, no. 2, pp. 555–567, jun 1992. [Online]. Available: http://doi.wiley.com/10.1002/pssa.2211310228

[91] R. Fisher, “Dispersion on a Sphere,” *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 217, no. 1130, pp. 295–305, may 1953. [Online]. Available: http://rspa.royalsocietypublishing.org/cgi/doi/10.1098/rspa.1953.0064

[92] J. T. Kent, “The Fisher-Bingham Distribution on the Sphere,” *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 44, no. 1, pp. 71–80, 1982. [Online]. Available: https://www.jstor.org/stable/2984712

[93] S. J. Julier and J. K. Uhlmann, “A New Extension of the Kalman Filter to Nonlinear Systems,” in *Proc. of AeroSense: The 11th Int. Symp. on Aerospace/Defense Sensing, Simulations and Controls*, vol. The 11th I, 1997, pp. 182–193.

[94] S. Kullback and R. A. Leibler, “On Information and Sufficiency,” *Ann. Math. Statist.*, vol. 22, no. 1, pp. 79–86, 1951.

[95] B. C. Hall, *Lie Groups, Lie Algebras, and Representations: An Elementary Introduction*, ser. Graduate Texts in Mathematics. Springer International Publishing, 2015. [Online]. Available: papers3://publication/uuid/5E1D254E-BB63-4484-B920-6A711F92B942

[96] C. E. Rasmussen, C. A. Uk, H. Nickisch, and H. M. De, “Gaussian Processes for Machine Learning (GPML) Toolbox,” *Journal of Machine Learning Research*, vol. 11, pp. 3011–3015, 2010.

[97] D. Henrich, C. Wurll, and H. Worn, “6 DOF path planning in dynamic environments-a parallel online approach,” in *Proceedings. 1998 IEEE International Conference on Robotics and Automation (Cat. No.98CH36146)*, vol. 1, no. May. IEEE, 1998, pp. 330–335. [Online]. Available: http://ieeexplore.ieee.org/document/676417/

[98] L. Kavraki, P. Svestka, J.-C. Latombe, and M. Overmars, “Probabilistic roadmaps for path planning in high-dimensional configuration spaces,” *IEEE Transactions on Robotics and Automation*, vol. 12, no. 4, pp. 566–580, 1996. [Online]. Available: http://ieeexplore.ieee.org/document/508439/

[99] J. J. Kuffner and S. M. Lavalle, “RRT-Connect An Efficient Approach to Single Query Path Planning,” *Robotics and Automation 2000*, vol. 54, no. 2, 2000. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=844730

[100] S. Van Vaerenbergh, M. Lazaro-Gredilla, and I. Santamaria, “Kernel Recursive Least-Squares Tracker for Time-Varying Regression,” *EEE Transactions on Neural Networks and Learning Systems*, vol. 23, no. 8, pp. 1313–1326, aug 2012.

[101] S. Choi, K. Lee, and S. Oh, “Robust learning from demonstration using leveraged Gaussian processes and sparse-constrained optimization,” in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, vol. 2016-June. IEEE, may 2016, pp. 470–475. [Online]. Available: http://ieeexplore.ieee.org/document/7487168/

[102] J. D. Mazimpaka and S. Timpf, “Trajectory data mining: A review of methods and applications,” *Journal of Spatial Information Science*, no. 13, dec 2016. [Online]. Available: http://josis.org/index.php/josis/article/view/263

[103] A. Bolbol, T. Cheng, I. Tsapakis, and J. Haworth, “Inferring hybrid transportation modes from sparse GPS data using a moving window SVM classification,” *Computers, Environment and Urban Systems*, vol. 36, no. 6, pp. 526–537, nov 2012. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0198971512000543

[104] Y. Zheng, Y. Chen, Q. Li, X. Xie, and W.-Y. Ma, “Understanding transportation modes based on GPS data for web applications,” *ACM Transactions on the Web*, vol. 4, no. 1, pp. 1–36, jan 2010. [Online]. Available: http://portal.acm.org/citation.cfm?doid=1658373.1658374

[105] J. C. Nascimento, M. Figueiredo, and J. S. Marques, “Trajectory Classification Using Switched Dynamical Hidden Markov Models,” *IEEE Transactions on Image Processing*, vol. 19, no. 5, pp. 1338–1348, may 2010. [Online]. Available: http://ieeexplore.ieee.org/document/5371914/

[106] J.-G. Lee, J. Han, X. Li, and H. Gonzalez, “TraClass: Trajectory classification using hierarchical region-based and trajectory-based clustering,” *Proceedings of the VLDB Endowment*, 2008.

[107] J.-G. Lee, J. Han, and X. Li, “A Unifying Framework of Mining Trajectory Patterns of Various Temporal Tightness,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 27, no. 6, pp. 1478–1490, jun 2015. [Online]. Available: http://ieeexplore.ieee.org/document/6977963/

[108] J. De Schutter, E. Di Lello, J. F. De Schutter, R. Matthysen, T. Benoit, and T. De Laet, “Recognition of 6 DOF rigid body motion trajectories using a coordinate-free representation,” in *2011 IEEE International Conference on Robotics and Automation*. IEEE, may 2011, pp. 2071–2078. [Online]. Available: http://ieeexplore.ieee.org/document/5980060/

[109] J. De Schutter, “Invariant Description of Rigid Body Motion Trajectories,” *Journal of Mechanisms and Robotics*, vol. 2, no. 1, p. 011004, 2010. [Online]. Available: https://people.mech.kuleuven.be/ jdeschut/ASME_JMR_2010_published.pdf

[110] T. L. Griffiths, C. G. Lucas, J. J. Williams, and M. L. Kalish, “Modeling human function learning with Gaussian processes,” in *Advances in Neural Information Processing Systems 21 - Proceedings of the 2008 Conference*, 2009, pp. 553–560. [Online]. Available: https://www.scopus.com/inward/record.uri?eid=2-s2.0-84863342363

[111] Y. Shen, N. Phan, X. Xiao, R. Jin, J. Sun, B. Piniewski, D. Kil, and D. Dou, “Dynamic socialized Gaussian process models for human behavior prediction in a health social network,” *Knowledge and Information Systems*, vol. 49, no. 2, pp. 455–479, nov 2016. [Online]. Available: http://link.springer.com/10.1007/s10115-015-0910-z

[112] Z. Wang, K. Mülling, M. P. Deisenroth, H. Ben Amor, D. Vogt, B. Schölkopf, and J. Peters, “Probabilistic movement modeling for intention inference in human–robot interaction,” *The International Journal of Robotics Research*, vol. 32, no. 7, pp. 841–858, jun 2013. [Online]. Available: http://journals.sagepub.com/doi/10.1177/0278364913478447

[113] L. Castellanos, V. Q. Vu, S. Perel, A. B. Schwartz, and R. E. . Kass, “A multivariate Gaussian process factor model for hand shape during reach-to-grasp movements,” *Statistica Sinica*, vol. 25, p. doi:10.5705/ss.2013.226w, 2015. [Online]. Available: http://www3.stat.sinica.edu.tw/statistica/J25N1/J25N11/J25N11.html

[114] H.-H. Bock, “Origins and extensions of the k-means algorithm in cluster analysis,” *Journal Electronique d'Histoire des Probabilités et de la Statistique*, vol. 4, no. 2, pp. 1–18, 2008.

[115] D. Comaniciu and P. Meer, “Mean shift: a robust approach toward feature space analysis,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 603–619, may 2002. [Online]. Available: http://ieeexplore.ieee.org/document/1000236/

[116] M. Ester, H. P. Kriegel, J. Sander, and X. Xu, “A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise,” *Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining*, pp. 226–231, 1996. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.121.9220

[117] R. Sibson, “SLINK: An optimally efficient algorithm for the single-link cluster method,” *The Computer Journal*, vol. 16, no. 1, pp. 30–34, jan 1973. [Online]. Available: https://academic.oup.com/comjnl/article-lookup/doi/10.1093/comjnl/16.1.30

[118] R. Cormack, “A Review of Classification,” *Journal of the Royal Statistical Society. Series A (General)*, vol. 134, no. 3, pp. 321–367, 1971.

[119] G. Schwarz, “Estimating the Dimension of a Model,” *The Annals of Statistics*, vol. 6, no. 2, pp. 461–464, mar 1978. [Online]. Available: http://projecteuclid.org/euclid.aos/1176344136

[120] A. Dempster, N. Laird, and D. B. Rubin, “Maximum likelihood from incomplete data via the EM algorithm,” *Journal of the Royal Statistical Society Series B Methodological*, vol. 39, no. 1, pp. 1–38, 1977. [Online]. Available: http://www.jstor.org/stable/10.2307/2984875

[121] S. Kim, R. Haschke, and H. Ritter, “Gaussian Mixture Model for 3-DoF orientations,” *Robotics and Autonomous Systems*, vol. 87, pp. 28–37, jan 2017. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0921889015302244

[122] F. L. Markley, Y. Cheng, J. L. Crassidis, and Y. Oshman, “Averaging Quaternions,” *Journal of Guidance, Control, and Dynamics*, vol. 30, no. 4, pp. 1193–1197, jul 2007. [Online]. Available: http://arc.aiaa.org/doi/10.2514/1.28949

[123] C. Robertson and K. A. Flowers, “Motor set in Parkinson’s disease.” *Journal of neurology, neurosurgery, and psychiatry*, vol. 53, no. 7, pp. 583–92, jul 1990. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/2391523

[124] L. M. de Lau and M. M. Breteler, “Epidemiology of Parkinson’s disease,” *The Lancet Neurology*, vol. 5, no. 6, pp. 525–535, jun 2006. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S1474442206704719

[125] D. Enders, M. Balzer-Geldsetzer, O. Riedel, R. Dodel, H.-U. Wittchen, S.-C. Sensken, B. Wolff, and J.-P. Reese, “Prevalence, Duration and Severity of Parkinson’s Disease in Germany: A Combined Meta-Analysis from Literature Data and Outpatient Samples,” *European Neurology*, vol. 78, no. 3-4, pp. 128–136, 2017. [Online]. Available: https://www.karger.com/Article/FullText/477165

[126] M. M. Hoehn and M. D. Yahr, “Parkinsonism: onset, progression and mortality.” *Neurology*, vol. 17, no. 5, pp. 427–42, may 1967. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/6067254

[127] N. Giladi, R. Kao, and S. Fahn, “Freezing phenomenon in patients with parkinsonian syndromes,” *Movement Disorders*, vol. 12, no. 3, pp. 302–305, may 1997. [Online]. Available: http://doi.wiley.com/10.1002/mds.870120307

[128] A. J. Hughes, S. E. Daniel, L. Kilford, and A. J. Lees, “Accuracy of clinical diagnosis of idiopathic Parkinson’s disease: a clinico-pathological study of 100 cases.” *Journal of Neurology, Neurosurgery & Psychiatry*, vol. 55, no. 3, pp. 181–184, mar 1992. [Online]. Available: http://jnnp.bmj.com/cgi/doi/10.1136/jnnp.55.3.181

[129] A. Berardelli, “Pathophysiology of bradykinesia in Parkinson’s disease,” *Brain*, vol. 124, no. 11, pp. 2131–2146, nov 2001. [Online]. Available: https://academic.oup.com/brain/article-lookup/doi/10.1093/brain/124.11.2131

[130] M. Hallett, “Parkinson revisited: pathophysiology of motor signs.” *Advances in neurology*, vol. 91, pp. 19–28, 2003. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/12442661

[131] G. Deuschl, U. Fietzek, S. Klebe, and J. Volkmann, “Chapter 24 Clinical neurophysiology and pathophysiology of Parkinsonian tremor,” in *Mov. Disord. Handb. Clin. Neurophysiol.*, H. M, Ed. Elsevier, 2003, ch. 24, pp. 377–396. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S1567423109701728

[132] C. G. Goetz, S. Fahn, P. Martinez-Martin, W. Poewe, C. Sampaio, G. T. Stebbins, M. B. Stern, B. C. Tilley, R. Dodel, B. Dubois, R. Holloway, J. Jankovic, J. Kulisevsky, A. E. Lang, A. Lees, S. Leurgans, P. A. LeWitt, D. Nyenhuis, C. W. Olanow, O. Rascol, A. Schrag, J. A. Teresi, J. J. Van Hilten, and N. LaPelle, “Movement Disorder Society-sponsored revision of the Unified Parkinson’s Disease Rating Scale (MDS-UPDRS): Process, format, and clinimetric testing plan,” *Movement Disorders*, vol. 22, no. 1, pp. 41–47, jan 2007. [Online]. Available: http://doi.wiley.com/10.1002/mds.21198

[133] P. Angeles, Y. Tai, N. Pavese, S. Wilson, and R. Vaidyanathan, “Automated assessment of symptom severity changes during deep brain stimulation (DBS) therapy for Parkinson’s disease,” in *2017 International Conference on Rehabilitation Robotics (ICORR)*. IEEE, jul 2017, pp. 1512–1517. [Online]. Available: http://ieeexplore.ieee.org/document/8009462/

[134] A. H. Butt, E. Rovini, C. Dolciotti, P. Bongioanni, G. De Petris, and F. Cavallo, “Leap motion evaluation for assessment of upper limb motor skills in Parkinson’s disease,” in *2017 International Conference on Rehabilitation Robotics (ICORR)*. IEEE, jul 2017, pp. 116–121. [Online]. Available: http://ieeexplore.ieee.org/document/8009232/

[135] A. Mirelman, I. Maidan, T. Herman, J. E. Deutsch, N. Giladi, and J. M. Hausdorff, “Virtual Reality for Gait Training: Can It Induce Motor Learning to Enhance Complex Walking and Reduce Fall Risk in Patients With Parkinson’s Disease?” *The Journals of Gerontology Series A: Biological Sciences and Medical Sciences*, vol. 66A, no. 2, pp. 234–240, feb 2011. [Online]. Available: https://academic.oup.com/biomedgerontology/article-lookup/doi/10.1093/gerona/glq201

[136] S. Papapetropoulos, H. L. Katzen, B. K. Scanlon, A. Guevara, C. Singer, and B. E. Levin, “Objective Quantification of Neuromotor Symptoms in Parkinson’s Disease: Implementation of a Portable, Computerized Measurement Tool,” *Parkinson’s Disease*, vol. 2010, pp. 1–6, 2010. [Online]. Available: http://www.hindawi.com/journals/pd/2010/760196/

[137] S. Summa, J. Tosi, F. Taffoni, L. Di Biase, M. Marano, A. C. Rizzo, M. Tombini, G. Di Pino, and D. Formica, “Assessing bradykinesia in Parkinson’s disease using gyroscope signals,” in *IEEE International Conference on Rehabilitation Robotics*, 2017, pp. 1556–1561.

[138] L. Tremblay, J. Esculier, J. Vaudrin, P. Bériault, and K. Gagnon, “Home-based balance training programme using Wii Fit with balance board for Parkinsons’s disease: A pilot study,” *Journal of Rehabilitation Medicine*, vol. 44, no. 2, pp. 144–150, 2012. [Online]. Available: http://www.medicaljournals.se/jrm/content/?doi=10.2340/16501977-0922

[139] B. M. Eskofier, S. I. Lee, J.-F. Daneault, F. N. Golabchi, G. Ferreira-Carvalho, G. Vergara-Diaz, S. Sapienza, G. Costante, J. Klucken, T. Kautz, and P. Bonato, “Recent machine learning advancements in sensor-based mobility analysis: Deep learning for Parkinson’s disease assessment,” in *2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, aug 2016, pp. 655–658. [Online]. Available: http://ieeexplore.ieee.org/document/7590787/

[140] N. H. Ghassemi, F. Marxreiter, C. F. Pasluosta, P. Kugler, J. Schlachetzki, A. Schramm, B. M. Eskofier, and J. Klucken, “Combined accelerometer and EMG analysis to differentiate essential tremor from Parkinson’s disease,” in *2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, aug 2016, pp. 672–675. [Online]. Available: http://ieeexplore.ieee.org/document/7590791/

[141] J. Cancela, M. Pansera, M. T. Arredondo, J. J. Estrada, M. Pastorino, L. Pastor-Sanz, and J. L. Villalar, “A comprehensive motor symptom monitoring and management system: The bradykinesia case,” *2010 Annual International Conference of the IEEE Engineering in Medicine and Biology*, vol. 2010, pp. 1008–1011, aug 2010. [Online]. Available: http://ieeexplore.ieee.org/document/5627775/

[142] S. H. Roy, B. T. Cole, L. D. Gilmore, C. J. De Luca, and S. H. Nawab, “Resolving signal complexities for ambulatory monitoring of motor function in Parkinson’s disease,” *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, vol. 2011, pp. 4836–4839, aug 2011. [Online]. Available: http://ieeexplore.ieee.org/document/6091198/

[143] F. Wahid, R. K. Begg, C. J. Hass, S. Halgamuge, and D. C. Ackland, “Classification of Parkinson’s Disease Gait Using Spatial-Temporal Gait Features,” *IEEE Journal of Biomedical and Health Informatics*, vol. 19, no. 6, pp. 1794–1802, nov 2015. [Online]. Available: http://ieeexplore.ieee.org/document/7138580/

[144] K. J. Kubota, J. A. Chen, and M. A. Little, “Machine learning for large-scale wearable sensor data in Parkinson’s disease: Concepts, promises, pitfalls, and futures,” *Movement Disorders*, vol. 31, no. 9, pp. 1314–1326, sep 2016. [Online]. Available: http://doi.wiley.com/10.1002/mds.26693

[145] D. G. M. Zwartjes, T. Heida, J. P. P. van Vugt, J. A. G. Geelen, and P. H. Veltink, “Ambulatory Monitoring of Activities and Motor Symptoms in Parkinson’s Disease,” *IEEE Transactions on Biomedical Engineering*, vol. 57, no. 11, pp. 2778–2786, nov 2010. [Online]. Available: http://ieeexplore.ieee.org/document/5462876/

[146] A. Filippeschi, N. Schmitz, M. Miezal, G. Bleser, E. Ruffaldi, and D. Stricker, “Survey of Motion Tracking Methods Based on Inertial Sensors: A Focus on Upper Limb Human Motion,” *Sensors*, vol. 17, no. 6, p. 1257, jun 2017. [Online]. Available: http://www.mdpi.com/1424-8220/17/6/1257

[147] J. I. Hoff, V. van der Meer, and J. J. van Hilten, “Accuracy of objective ambulatory accelerometry in detecting motor complications in patients with Parkinson disease.” *Clinical neuropharmacology*, vol. 27, no. 2, pp. 53–7, 2004. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/15252264

[148] N. L. Keijsers, M. W. Horstink, and S. C. Gielen, “Ambulatory motor assessment in Parkinson’s disease,” *Movement Disorders*, vol. 21, no. 1, pp. 34–44, jan 2006. [Online]. Available: http://doi.wiley.com/10.1002/mds.20633

[149] C. Pérez-López, A. Samà, D. Rodríguez-Martín, A. Català, J. Cabestany, J. Moreno-Arostegui, E. de Mingo, and A. Rodríguez-Molinero, “Assessing Motor Fluctuations in Parkinson’s Disease Patients Based on a Single Inertial Sensor,” *Sensors*, vol. 16, no. 12, p. 2132, 2016. [Online]. Available: http://www.mdpi.com/1424-8220/16/12/2132

[150] M. D. Hssayeni, M. A. Burack, and B. Ghoraani, “Automatic assessment of medication states of patients with Parkinson’s disease using wearable sensors,” in

2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC). IEEE, aug 2016, pp. 6082–6085. [Online]. Available: http://ieeexplore.ieee.org/document/7592116/

[151] A. Tzallas, M. Tsipouras, G. Rigas, D. Tsalikakis, E. Karvounis, M. Chondrogiorgi, F. Psomadellis, J. Cancela, M. Pastorino, M. Waldmeyer, S. Konitsiotis, and D. Fotiadis, "PERFORM: A System for Monitoring, Assessment and Management of Patients with Parkinson's Disease," *Sensors*, vol. 14, no. 11, pp. 21 329–21 357, nov 2014. [Online]. Available: http://www.mdpi.com/1424-8220/14/11/21329/

[152] R. W. Gibb and A. J. Lees, "The Significance of the Lewy Body in the Diagnosis of Idiopathic Parkinson's Disease," *Neuropathology and Applied Neurobiology*, vol. 15, no. 1, pp. 27–44, feb 1989. [Online]. Available: http://doi.wiley.com/10.1111/j.1365-2990.1989.tb01147.x

[153] R. I. Griffiths, K. Kotschet, S. Arfon, Z. M. Xu, W. Johnson, J. Drago, A. Evans, P. Kempster, S. Raghav, and M. K. Horne, "Automated Assessment of Bradykinesia and Dyskinesia in Parkinson's Disease." *Journal of Parkinson's disease*, vol. 2, no. 1, pp. 47–55, 2012. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/23939408

[154] C. Ossig, F. Gandor, M. Fauser, C. Bosredon, L. Churilov, H. Reichmann, M. K. Horne, G. Ebersbach, and A. Storch, "Correlation of Quantitative Motor State Assessment Using a Kinetograph and Patient Diaries in Advanced PD: Data from an Observational Study," *PLOS ONE*, vol. 11, no. 8, p. e0161559, aug 2016. [Online]. Available: https://doi.org/10.1371/journal.pone.0161559