

Electrocardiogram Signal Analysis and Simulations for Non-Invasive Diagnosis

Model-Based and Data-Driven Approaches for the Estimation
of Ionic Concentrations and Localization of Excitation Origins

Zur Erlangung des akademischen Grades eines

DOKTOR-INGENIEURS

von der KIT-Fakultät für

Elektrotechnik und Informationstechnik

des Karlsruher Instituts für Technologie (KIT)

genehmigte

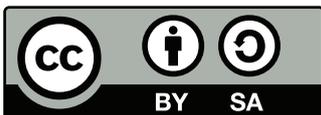
DISSERTATION

von

Nicolas Alessandro Pilia, M.Sc.

geb. in Wiesbaden

Tag der mündlichen Prüfung:	18.02.2021
Referent:	Prof. Dr. rer. nat. Olaf Dössel
Korreferentin:	ass. Prof. Dr. Cristiana Corsi



*This document - excluding the cover, pictures, tables and graphs - is licensed under the Creative Commons Attribution-ShareAlike 4.0 International License (CC BY-SA 4.0): <https://creativecommons.org/licenses/by-sa/4.0/>
A possible CC license of pictures, tables and graphs is stated in the respective captions.*

Abstract

The electrocardiogram (ECG) is the standard measurement device of the electrical heart activity. It is highly available and allows for a quick, inexpensive, and non-invasive monitoring. This is especially important for the diagnosis of cardiovascular disease (CVD) which is one of the major concerns for the health care system in Europe. CVD causes costs of €210 billion and is responsible for 3.9 million deaths (45% of all deaths) a year. Apart from risk factors, chronic kidney disease (CKD) and structural changes in the heart tissue are underlying pathologies causing CVD. Both diseases can lead to life-threatening arrhythmia. This is why the following two pathologies connected to CVD are focused on in this thesis: Electrolyte imbalances in CKD patients and ectopic foci in the ventricles autonomously triggering an excitation. In both cases, the overall goal is to develop methods with the help of simulated signals supporting diagnosis.

In the first project, ECG simulations are used to optimize a signal processing workflow for an ECG-based estimation of blood potassium concentration ($[K^+]_b$) and blood calcium concentration ($[Ca^{2+}]_b$). The findings from the simulation studies are incorporated into two $[K^+]_b$ estimation methods which are evaluated on patient data. Mean absolute estimation errors were 0.37 mmol/l for a patient-specific approach and 0.48 mmol/l for a global approach with patient-specific adjustment. Advantages compared to existing approaches are extensively discussed. All algorithms being important for a signal processing workflow are published under an open source license.

The second project aims at estimating the location of ectopic foci with the surface ECG without knowing the individual geometry of the patient. 1,766,406 simulated ECG signals (body surface potential maps (BSPMs)) are utilized to train two convolutional neural networks (CNNs): The first estimates start and end of the depolarization, the second uses the depolarization part in the BSPM to localize the excitation origin. This CNN is designed to be able to show multiple solutions in the case of several possible excitation origins. The smallest median localization errors were 1.54 mm on the test set for the simulated and 37 mm for the patient data. Hence, the combination of the two CNNs yields a reliable method for the localization of ectopic foci on simulated and on patient data, although patient signals were not used during training.

The results from the two projects demonstrate how simulated data can be used to develop and improve adequate ECG signal processing methods and how diagnosis can be supported. Furthermore, the potential of the combination of simulations and CNNs for overcoming the problem of unavailable clinical datasets as well as for finding estimation models being

valid for different patients is demonstrated. The proposed methods can be used to accelerate diagnosis and is therefore likely to improve the outcome of the patients.

Zusammenfassung

Das Elektrokardiogramm (EKG) ist die Standardtechnik zur Messung der elektrischen Aktivität des Herzens. EKG-Geräte sind verfügbar, kostengünstig und erlauben zudem eine nichtinvasive Messung. Das ist insbesondere wichtig für die Diagnose von kardiovaskulären Erkrankungen (KVE). Letztere sind mit verursachten Kosten von 210 Milliarden Euro eine der Hauptbelastungen für das Gesundheitssystem in Europa und dort der Grund für 3,9 Millionen Todesfälle – dies entspricht 45% aller Todesfälle. Neben weiteren Risikofaktoren spielen chronische Nierenerkrankungen und strukturelle Veränderungen des Herzgewebes eine entscheidende Rolle für das Auftreten von KVE. Deshalb werden in dieser Arbeit zwei Pathologien, die in Verbindung zu KVE stehen, betrachtet: Elektrolytkonzentrationsveränderungen bei chronisch Nierenkranken und ektope Foki, die autonom Erregungen initiieren. In beiden Projekten ist die Entwicklung von Methoden mithilfe von simulierten Signalen zur Diagnoseunterstützung das übergeordnete Ziel.

Im ersten Projekt helfen simulierte EKGs die Signalverarbeitungskette zur EKG-basierten Schätzung der Ionenkonzentrationen von Kalium und Calcium zu optimieren. Die Erkenntnisse dieser Optimierung fließen in zwei patienten-spezifische Methoden zur Kaliumkonzentrations-schätzung ein, die wiederum mithilfe von Patientendaten ausgewertet werden. Die Methoden lieferten im Mittel einen absoluten Fehler von 0,37 mmol/l für einen patienten-spezifischen Ansatz und 0,48 mmol/l für einen globalen Ansatz mit zusätzlicher patienten-spezifischer Korrektur. Die Vorteile der Schätzmethoden werden gegenüber bereits existierenden Ansätzen dargelegt. Alle entwickelten Algorithmen sind ferner unter einer Open-Source-Lizenz veröffentlicht.

Das zweite Projekt zielte auf die Lokalisierung von ektope Foki mithilfe des EKGs ohne die Nutzung der individuellen Patientengeometrie. 1.766.406 simulierte EKG-Signale (Body Surface Potential Maps (BSPMs)) wurden zum Trainieren von zwei Convolutional Neural Networks (CNNs) erzeugt. Das erste CNN sorgt für die Schätzung von Anfang und Ende der Depolarisation der Ventrikel. Das zweite CNN nutzt die Information der Depolarisation im BSPM zur Schätzung des Erregungsursprungs. Der spezielle Aufbau des CNNs ermöglicht die Darstellung mehrerer Lösungen, wie sie durch Mehrdeutigkeiten im BSPM vorliegen können. Der kleinste Median des Lokalisierungsfehlers lag bei 1,54 mm für den Test-Datensatz der simulierten Signale, bzw. bei 37 mm für Patientensignale. Somit erlaubt die Kombination beider CNNs die verlässliche Lokalisierung von ektope Foki auch anhand von Patientendaten, obwohl Patientendaten vorher nicht im Training genutzt wurden.

Die Resultate dieser zwei Projekte demonstrieren, wie EKG-Simulationen zur Entwicklung und Verbesserung von EKG-Signalverarbeitungsmethoden eingesetzt werden und bei der Diagnosefindung helfen können. Zudem zeigt sich das Potential der Kombination von Simulationen und CNNs, um einerseits die zumeist raren klinischen Signale zu ersetzen und andererseits Modelle zu finden, die für mehrere Patienten/-innen gültig sind. Die vorgestellten Methoden bergen die Möglichkeit, die Diagnosestellungen zu beschleunigen und mit hoher Wahrscheinlichkeit den Therapieerfolg der Patienten zu verbessern.

Acknowledgments

This thesis is the final result of my scientific work at the Institute of Biomedical Engineering at Karlsruhe Institute of Technology (KIT) during the last years. Many people contributed to the finalization of work and I would like to sincerely thank them in the following.

First, I want to thank Prof. Dr. rer. nat. Olaf Dössel for letting me join his research group, for his scientific enthusiasm, his support until the finalization of this thesis and for always having my back. Moreover, I would like to thank ass. Prof. Cristiana Corsi for providing me important data for my research, for refereeing this thesis and for always being such a kind and helpful person. The same holds for Dr. Stefano Severi who mainly hosted our research stay in Cesena, which I will always remember as a wonderful time, and gave me and my student valuable feedback for the work. I also want to thank Dr. Axel Loewe for the fruitful collaboration, his excellent feedback on our joint publications, and his support during my work. This embraces the contact to Peter Kotanko and Dr. Jochen Reimann from Renal Research Institute, as well as Dr. Doris Fürtinger from Fresenius Medical Care who kindly supported the student research project of María Hernández Mesa financially.

All my colleagues and former colleagues at the institute deserve a special thank. In particular, I would like to thank Steffen Schuler for proof-reading the parts of the first draft of this thesis, the pleasant collaboration where we strongly benefit from each other. I appreciate him as one of the most helpful colleagues at the institute and especially as a good friend. Furthermore, I would like to thank Laura Unger for the pleasant work in the joint projects regarding intracardiac processing, for her time spent on correcting this thesis, and for all the various talks and AcroYoga sessions which helped to re-focus on work. Thanks for being such a sympathetic and helpful friend. I would like to thank Claudia Nagel for being a pleasant office mate, for the discussions about ECG processing and simulations, for the support with selected student projects, but also for her effort on revising the draft of this thesis. In addition, I thank Andreas Wachter for being always helpful, for the nice scientific and management discussions with him and Steffen Schuler at Vogelbräu, Ady Naber for always being available for a short talk in his office and his visits on the third floor, Michael Kircher for the support with my teaching task and for the constructive conversations about management, Yannick Lutz for making my time at the institute hilarious for about seven years, Ekaterina Kovacheva for being such a positive person and for reminding me to continue smiling, Mark Nothstein for always having a compliment for everyone, Giorgio Luongo for the nice cooperation and being a large-minded person, Jochen Brenneisen, Luca Azzolin, Jorge Sánchez and Deborah Nairn for their support with my teaching task. I would like to thank Tiago Almeida for the

countless discussions about research, about life in general and about Whisky during his stay at the institute. Furthermore, Dr. Stefan Pollnow for a fruitful collaboration and Dr. Gustavo Lenis for introducing me into the research world. I want to express a heartfelt thank you also to all the others I did not mention here: You form an excellent team. Please preserve this atmosphere.

Apart from the scientific staff, I would like to thank the technical and administrative staff in the following who supported me in organizing and teaching the class Lineare Elektrische Netze for three years: Ich danke Ramona Modery, die mich während der drei Jahre, in denen ich die Lehrveranstaltung Lineare Elektrische Netze organisierte und die Übung leitete, enorm und verlässlich unterstützte. Das gleiche gilt für die Sekretärin des Instituts, Irene Günter, die gewissenhaft alle Verwaltungsaufgaben, insbesondere die Eintragung der Noten, übernommen hat. Weiterhin danke ich Manfred Schroll für seine Unterstützung in Sachen IT-Problemen. Wenn es darauf ankam, war er immer da. Ich sowie meine Studenten konnten uns immer seiner Hilfe sicher sein.

My Bachelor and Master students deserve in particular my severe gratitude: Gerald Moik, Michael Meinzer, Silvia Becker, Moritz Lindner, Jeanne Koch, Claudia Nagel, Bilian Smardanski, and Simon Süß. Special thanks go to María Hernández Mesa and Maike Rees, who supported me until the end of my work. As well, I thank all of my tutors for Lineare Elektrische Netze. The nice teamwork with all of the students helped me to recognize my enthusiasm to lead small teams.

I thank all of my friends for being patient with me, for always being there for me and supporting me. Especially, I want to thank Alexander Engelmann for his valuable feedback on optimization topics and Lisa Sielaff for the best and deepest feedback on texts I have ever got.

Furthermore, I want to thank my girlfriend Lorena for always having an open ear and encouraging words for me. A last thank goes to my family who unconditionally backed me: Danke Mama, Grazie Papà, Danke Marisa für eure Unterstützung und Geduld mit mir. Ich bin überglücklich, euch zu haben.

Contents

Abstract	i
Zusammenfassung	iii
Acknowledgments	v
Abbreviations	xi
1 Introduction	1
1.1 Motivation	1
1.2 Objectives of the Thesis	2
1.3 Structure of the Thesis	3
<hr/>	
I Fundamentals	5
<hr/>	
2 Medical Fundamentals	7
2.1 Physiology and Anatomy of the Heart	7
2.2 Electrocardiography	10
2.3 Disturbances of the Ionic Concentration Homeostasis	11
2.4 Ectopic Sources and Ventricular Tachycardia	14
3 Technical Fundamentals	15
3.1 Electrophysiological Modeling	15
3.2 Regression and Classification Methods	19
<hr/>	
II Ionic Concentration Estimation with the ECG	31
<hr/>	
4 Introduction to the Topic	33
5 Simulation Studies	35
5.1 Himeno et al. Whole Heart Simulations	35
5.2 Optimization of Pre-processing Steps	38
5.3 Proof of Concept Study for Concentration Estimation	51
5.4 Influence of Patient Geometry	53

5.5	Conclusions from the Simulation Studies	62
6	Application to Clinical Data	63
6.1	Introduction	63
6.2	Dataset	64
6.3	Methods	64
6.4	Results	66
6.5	Discussion	67
6.6	Summary and Outlook	74
7	Conclusion	77
<hr/>		
III	Localization of Ectopic Foci with the ECG	79
<hr/>		
8	Introduction to the Topic	81
9	Proof of Concept Study	83
9.1	Motivation	83
9.2	Simulations	83
9.3	Methods	84
9.4	Results	85
9.5	Discussion	87
9.6	Conclusion	88
9.7	Outlook	88
10	Deep Learning Approach	89
10.1	Introduction	89
10.2	Simulations	89
10.3	Methods	93
10.4	Results	98
10.5	Application to Clinical Data	100
10.6	Discussion	105
10.7	Outlook	108
11	Conclusion	111
<hr/>		
IV	Open Source Projects	113
<hr/>		
12	Introduction to the Topic	115
13	ECGdeli - ECG Delineation Algorithms	117
13.1	Introduction	117
13.2	Performance	118

13.3 Impact	118
14 ECGfeat - ECG Feature Extraction Algorithms	121
14.1 Introduction	121
14.2 Contents	122
14.3 Pre-processing Recommendation and Robustness Evaluation	122
14.4 Impact	124
15 ECGconc - ECG Concentration Estimation Algorithms	127
15.1 Introduction	127
15.2 Contents	127
15.3 Impact	128
16 Summary and Conclusion	129
<hr/>	
V Final Remarks	131
<hr/>	
17 Conclusion	133
18 Ideas for Future Projects	135
18.1 Machine Learning and ECG Signal Processing	135
18.2 Intracardiac Signal Processing	137
A Appendix	139
A.1 Himeno et al. Whole Heart Simulations	140
A.2 Influence of Filtering on Feature Extraction	141
A.3 Influence of Patient Geometry	145
A.4 ECGdeli - ECG Delineation Algorithms	147
A.5 Convolutional Neural Network Structures	148
A.6 ScaleNet Error Histograms	160
A.7 LocaNet Error Histograms	162
A.8 Results on Clinical Data	165
References	167
List of Publications and Supervised Theses	179

Abbreviations

ANN	artificial neural network
AP	action potential
AV	atrioventricular
BSP	body surface potential
BSPM	body surface potential map
[Ca²⁺]_o	extracellular calcium concentration
[Ca²⁺]_b	blood calcium concentration
Ca²⁺	calcium
CCA	Canonical Correlation Analysis
CKD	chronic kidney disease
Cl⁻	chloride
CNN	convolutional neural network
CVD	cardiovascular disease
ECG	electrocardiogram
ECGI	ECG imaging
K⁺	potassium
[K⁺]_o	extracellular potassium concentration
[K⁺]_b	blood potassium concentration
MLP	multilayer perceptron
Na⁺	sodium
PCA	principal component analysis
SCD	sudden cardiac death
SNR	signal to noise ratio
SVR	support vector regression
TMV	transmembrane voltage

Introduction

1.1 Motivation

Cardiovascular diseases (CVDs) are one of the major concerns for the healthcare system being responsible for over 3.9 million deaths a year, or 45% of deaths in Europe (40% in men, 49% in women) [1]. In contrast, cancer accounts for only under 1.1 million deaths. Among CVD, heart ischemia and infarction are the single most cause of death and responsible for 862,000 deaths a year. These numbers reflect also in the expenses: it is estimated that CVD cause costs of €210 billion a year in Europe. The exact cause for CVD can be various but it is known that secondary prevention, i.e. the therapy of other underlying diseases, reduce the risk of CVD events [2]. Apart from classical risk factors like smoking, diabetes, high blood pressure, or obesity, chronic kidney disease (CKD) and structural changes in the heart tissue (e.g. as a result of myocardial infarction) are further underlying problems causing CVD [2–4]. The two latter pathologies can lead to life-threatening arrhythmia and can become therefore the original cause for the disease [5–11]. An early diagnosis and therapy of these causal diseases can help to improve the outcome of the patients [12–14].

The electrocardiogram (ECG) is the standard monitoring tool of the electrical activity of the heart over time. It is available and used in nearly every clinical setting. This makes this device highly available and allows for a quick inexpensive non-invasive measurement of the electrical heart activity. Subsequently, the ECG is applied during the first diagnosis steps in CVD patients. ECG data can be used to calculate rhythmical and morphological features in which the physician decides about the optimal treatment for the patient. Apart from the clinical setting, ECG measurement devices become more and more included into consumable devices, e.g. the Apple Watch. A trend towards transferring the evaluation of the health status of the user to an ECG system exists [15, 16]. The automatic detection and evaluation of heart diseases has the potential to support a quick and early in- and out-of-hospital diagnosis and support physicians in decision making during treatment.

As the automatic determination of the health status of the patient affects therapy decisions and therefore patient outcomes, algorithms have to be designed carefully: All possible influences on the result must be evaluated under the constraint that medical data show large inter-patient

variations and are not easily accessible due to privacy regulations and an infrastructure in the clinics that is not always capable of exporting the needed digital data. One possibility to overcome these constraints are simulations of ECGs. Computational modeling of the heart offers the possibility to create as many data as desired being able to control all influences separately. This can help to gain knowledge about the problem and match the evaluation algorithms to the findings. Nevertheless, computational modeling alone can only be the first or a supporting step towards a clinical application. Every method has to be proven and evaluated on clinical data which usually show larger temporal and individual variations than simulations.

In this thesis, two pathologies were focused on which can both lead to life-threatening arrhythmia: Electrolyte imbalances in CKD patients and ectopic foci in the ventricles autonomously triggering an excitation. In both projects, simulations were utilized to develop methods being applicable to patient data and support in the detection and curing of these diseases.

1.2 Objectives of the Thesis

The general aim of this work is to develop ECG signal processing methods to extract meta information from the ECG. These can be used to accelerate the diagnostic workflow by supporting physicians finding diagnoses and improving the patient's outcome.

As already stated, two major topics will be incorporated into this thesis: The estimation of electrolyte concentrations and the localization of ectopic foci in the ventricles. The leading aims of the first project are:

1. Improve ECG feature extraction methods with regard to a clinical application.
2. Perform a proof-of-concept study for the ECG-based blood potassium concentration ($[K^+]_b$) and blood calcium concentration ($[Ca^{2+}]_b$) estimation.
3. Evaluate the influence of patient-specific anatomical properties on feature extraction and therefore concentration estimation.
4. Develop a method for ECG-based $[K^+]_b$ estimation using findings from the first three studies and overcoming drawbacks from published approaches.

The second topic described in this thesis is the localization of ectopic foci. The aims of this project comprised:

1. Implement a method to detect the start and end of the depolarization wave to be used in the localization procedure.
2. Perform simulations to train a localization technique that can be also applied to clinical data.
3. Localize the excitation origin in the ventricles on both simulated and clinical data as precise as possible.
4. Visualize the results and possible ambiguities.

1.3 Structure of the Thesis

Part I presents selected fundamentals being relevant for understanding the presented methods and results:

- **Chapter 2** contains the medical background being important for this work. Anatomical and physiological phenomena are explained as well as the origin of the ECG.
- **Chapter 3** presents an overview on the technical fundamentals. An introduction to computational modeling of the heart is given. Moreover, regression and classification techniques pertinent for this work are presented.

Part II comprises the concentration estimation project using model based and data driven methods.

- **Chapter 4** gives a general introduction to the topic.
- **Chapter 5** contains the projects connected with simulations. The main goals were the optimization of pre-processing steps, the evaluation of possible inter-patient variations and a proof-of-concept study for $[K^+]_b$ and $[Ca^{2+}]_b$ estimation.
- **Chapter 6** outlines two approaches for concentration estimation applied to clinical data that are based on the findings from the simulation studies.
- **Chapter 7** gives a general conclusion of this part.

Part III describes methods for ectopic foci localization.

- **Chapter 8** comprises a general introduction into the field of ectopic foci localization.
- **Chapter 9** presents results from a proof-of-concept study based on simulations and support vector regression evaluating the feasibility of an ECG-based localization approach.
- **Chapter 10** studies the application of a deep learning approach for the localization of ectopic foci.
- **Chapter 11** presents a conclusion of this part.

Part IV describes three open source projects containing algorithms developed during this thesis.

- **Chapter 12** introduces and motivates the topic of open source research software.
- **Chapter 13** presents ECGdeli, a highly modular collection of ECG pre-processing and delineation algorithms being important for most studies in this thesis.
- **Chapter 14** outlines ECGfeat, the first collection of ECG feature extraction algorithms on github that were evaluated and optimized regarding their robustness.
- **Chapter 15** describes the repository ECGconc containing all algorithms being important for concentration estimation.
- **Chapter 16** outlines a summary and conclusion of this part.

Part V summarizes the findings from all of the studies from a general perspective and gives a comment on topics worth to work on. This includes the projects that I worked on during my time at the Institute of Biomedical Engineering and are not reported in this thesis. The topics were mostly connected with intracardiac signal processing, comprising the estimation of the conduction velocity in the atria and the quantification of the complexity of atrial signals. Selected results obtained during my time at the Institute of Biomedical Engineering were published in five journal papers out of which three are with a shared authorship. Two further publications are in preparation and writing has already started. The articles are mostly referenced at the beginning of the respective sections and are listed again at the end of this thesis. Furthermore, I supervised 14 student projects. Findings from these projects were partly considered during development and assessment of new methods and results were incorporated into this thesis. The respective theses are also referenced at the beginning of the respective sections.

PART I

FUNDAMENTALS

Medical Fundamentals

2.1 Physiology and Anatomy of the Heart

The human heart is a hollow muscle enclosed by the pericardial sac which is located in the mediastinum, the space between spine and sternum. The heart consists of four cavities: two atria and two ventricles (Figure 2.1) connected to each other either directly or through the circulatory system. Starting from the venae cavae through which low-oxygenated blood flows into the right atrium, the blood flow is driven by contraction and by other filling mechanisms out of the right atrium into the right ventricle passing the tricuspid valve. When the right ventricle contracts, the blood is pumped into the pulmonary vessel system where it is oxygenated and returns into the left atrium. From there, the blood moves into the left ventricle and is pushed into the aorta and thereby into the body circulatory system. From here, the oxygen is released into the cells, and the blood flow returns through the venae cavae into the right atrium [17, pp. 522-525].

2.1.1 Conduction System of the Heart

The trigger for the heart muscle cells to contract arrives the cells through the conduction system of the heart (Figure 2.2). Starting from the sinus node, the primary pacemaker of the heart located in the right atrium, the excitation is conducted over the atria into the second pacemaker, the atrioventricular (AV) node. Atria and ventricles are electrically isolated and a propagation of the excitation wave is only possible through the AV node. From there, the excitation wave is conducted with a delay through the bundle of His (third pacemaker), over Tawara's branches and the Purkinje fibers into the working myocardium. The latter is formed of contractile cells being responsible for most of the contraction. The cells forming all the pacemakers are self-depolarizing, i.e. after a certain time, they can provoke a new excitation of the heart. The frequency of spontaneous depolarization descends from the primary to the fourth pacemaker. This results in a domination of the sinus node fixing the rate of contraction (60 to 80 bpm) in the physiological case. If the primary pacemaker fails

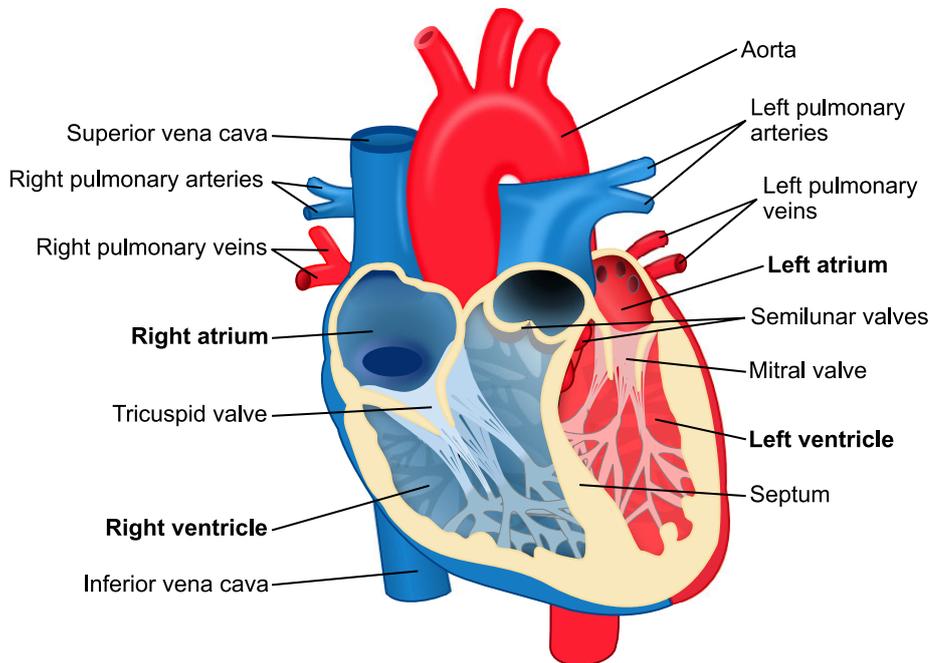


Figure 2.1: Anatomy of the heart. Modified from [18], licensed under Creative Commons Attribution-Share Alike 3.0 Unported.

to trigger the excitation, the second (40 to 50 bpm) pacemaker, the AV node can act as a fall-back. Equivalently, the third pacemaker (30 to 40 bpm) compensates a failure of the second pacemaker [17, pp. 522-525].

2.1.2 Electrophysiology

The excitation of a heart cell is determined by the change of the transmembrane voltage (TMV). When a neighboring cell is depolarized, a small charge transfer is induced which triggers the cell to change its TMV as shown in Figure 2.3. This typical course is referred to as action potential (AP). When hitting the threshold potential (-70 mV), certain inward (into the cell) potassium (K^+) channels that are open during the resting phase get deactivated and a fast sodium (Na^+) inward current increases the TMV up to 20- 40 mV (upstroke phase 0 and overshoot phase 1 in Figure 2.3). Afterwards, the fast Na^+ channels deactivate dependent on the TMV. Repolarizing K^+ outward and chloride (Cl^-) inward currents make the TMV decrease again. In the plateau phase (phase 2 in Figure 2.3), slow calcium (Ca^{2+}) inward currents compensate the repolarizing currents holding the TMV on a nearly constant level. Dependent on the heart rate and the cell location, the Ca^{2+} inward currents last for 200-400 ms determining also the length of the plateau. With the decay of these Ca^{2+} currents, the cell activation enters the repolarization phase (phase 3 in Figure 2.3). Mainly slow and

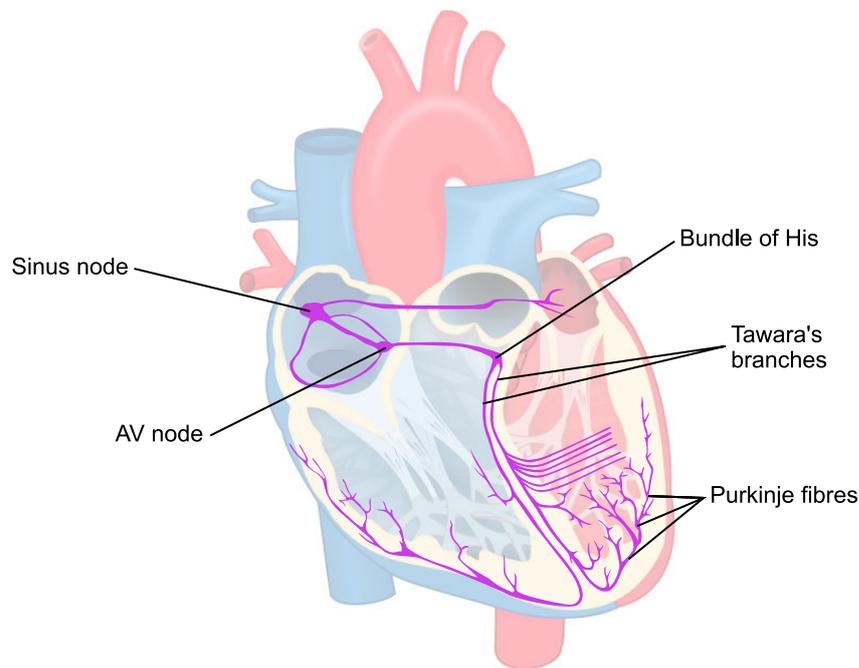


Figure 2.2: Anatomy of the heart and conduction system of the heart. Modified from [18] and from [19], both licensed under Creative Commons Attribution-Share Alike 3.0 Unported.

rapid K^+ outward currents drive the TMV back to the resting potential. There, the rapid and slow K^+ outward channels close and certain inward K^+ channels open again that are enabled during the resting phase (phase 4 in Figure 2.3). Here, the cell can be activated again, whereas during the plateau phase, the cell is in the absolute refractory period, i.e. a new activation is not possible. During the repolarization phase, the cell is relatively refractory, i.e. with decreasing TMV the needed trigger potential decreases, too [17, pp. 518-520]. Apart from the ion channels, there are other transporting mechanisms for ions through the cell membrane. Two important are: the Na^+ pump and the Na^+ - Ca^{2+} exchanger. Those types of transporters restore the original distribution of K^+ , Na^+ and Ca^{2+} ions between intra- and extracellular space [17, p. 520].

2.1.3 Electromechanical Coupling

As explained in Section 2.1.2, the Ca^{2+} influx sustains the plateau phase. Apart from this electrical effect, the increased concentration of Ca^{2+} triggers an additional release of Ca^{2+} from the sarcoplasmic reticulum in the cell into the cytosol. This Ca^{2+} activates the proteins of the contractile apparatus and therefore the mechanical activation of the cell [17, p. 521].

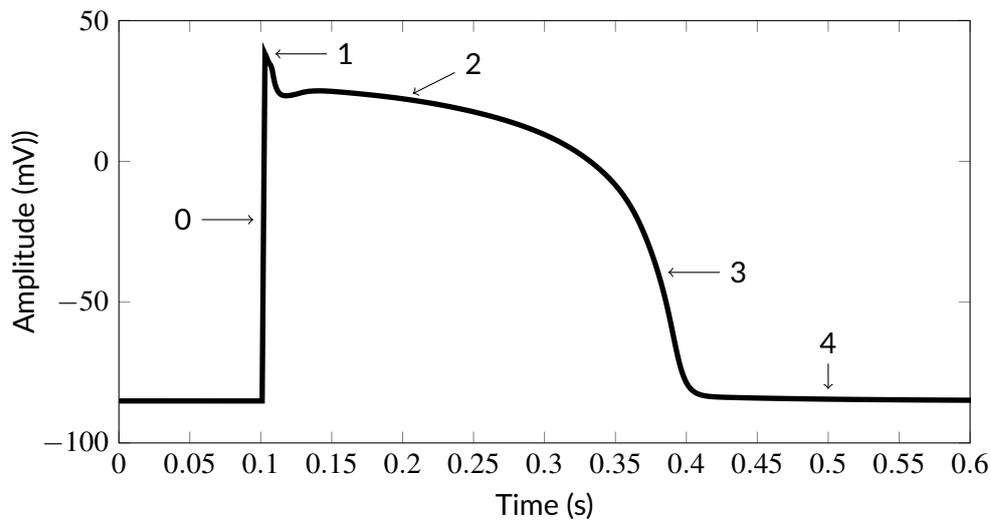


Figure 2.3: AP from a heart muscle cell with the respective phases. Phase 0: Upstroke/Depolarization, fast Na^+ influx; Phase 1: Overshoot, Na^+ channels close; Phase 2: Plateau, Ca^{2+} influx; Phase 3: Repolarization, K^+ outflux; Phase 4: Resting membrane potential.

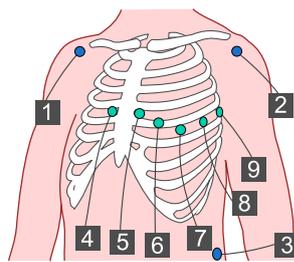


Figure 2.4: Placement of electrodes for a standard 12-lead ECG on the torso. Electrodes 1-3 are used to measure the Einthoven and Goldberger leads, as well as to calculate Wilson central terminal. Electrodes 4-9 are used to measure the Wilson leads together with Wilson central terminal. Modified from [20], licensed under Creative Commons Attribution-Share Alike 3.0 Unported.

2.2 Electrocardiography

Electrocardiography is one of the most common non-invasive diagnostic tools for heart diseases. As it is complicated to evaluate the health status of the heart cells within the body, the electrocardiogram (ECG) can be used to get a macroscopic view on the heart's status. Therefore, different voltages are measured with electrodes on the torso surface resulting in characteristic curves that can be interpreted by a physician.

Figure 2.4 shows the typical placement of the electrodes for a standard 12-lead ECG on the torso. The 12-lead ECG consists of three Einthoven leads (voltages I, II, III), three Goldberger leads (voltages aVR, aVL, aVF) and six Wilson leads (voltages V1-V6). Although there are 12 voltages to be measured, only nine electrodes suffice for an exact measurement. This is

due to the fact, that one Einthoven lead and all three Goldberger leads are mathematically redundant. Einthoven leads I and II can be calculated from the potentials at the two arms (Φ_{RA} , Φ_{LA}) and the left leg (Φ_{LL}) using

$$I = \Phi_{RA} - \Phi_{LA} , \quad (2.1)$$

$$II = \Phi_{RA} - \Phi_{LL} . \quad (2.2)$$

$$(2.3)$$

The redundant leads can be calculated from the others as follows [21]

$$III = -I + II \quad (2.4)$$

$$aVR = -\frac{I + II}{2} , \quad (2.5)$$

$$aVL = \frac{I - III}{2} , \quad (2.6)$$

$$aVF = \frac{II + III}{2} . \quad (2.7)$$

The voltages of the Wilson leads usually use the reference potential of Wilson central terminal Φ_{WCT} [22]

$$\Phi_{WCT} = \frac{\Phi_{LA} + \Phi_{RA} + \Phi_{LL}}{3} , \quad (2.8)$$

which is used to calculate the precordial leads [22] with

$$V_n = \Phi_{Vn} - \Phi_{WCT} \text{ with } n = 1, \dots, 6 . \quad (2.9)$$

The characteristic lead signals originating from one excitation cycle are shown in Figure 2.5. When the excitation spreads over the atria, the P wave gets visible in the ECG. Afterwards, the excitation is delayed in the AV node resulting in the PQ interval and then conducted into the ventricles resulting in their depolarization, represented by the QRS complex, and their repolarization, represented by the T wave [23, pp. 12-13]. As the repolarization in each cell is much slower than the depolarization, the resulting spatial gradient should also be much smaller. Nevertheless, the T wave is more prominent than expected. This is connected to regional repolarization profiles yielding characteristic spatial gradients that are captured with the ECG [24].

2.3 Disturbances of the Ionic Concentration Homeostasis

The physiological concentrations of Na^+ , K^+ and Ca^{2+} in the blood plasma are 141 mmol/l, 4 mmol/l and 2.5 mmol/l, respectively. In the interstitial space, the concentration of Ca^{2+} differs (1.3 mmol/l) [17, p. 669]. The homeostasis of the ionic concentration is mostly preserved by the kidneys [17, p. 665]. A (chronic) dysfunction can evoke pathological concentrations in the blood. However, not only kidney diseases can cause changed ion

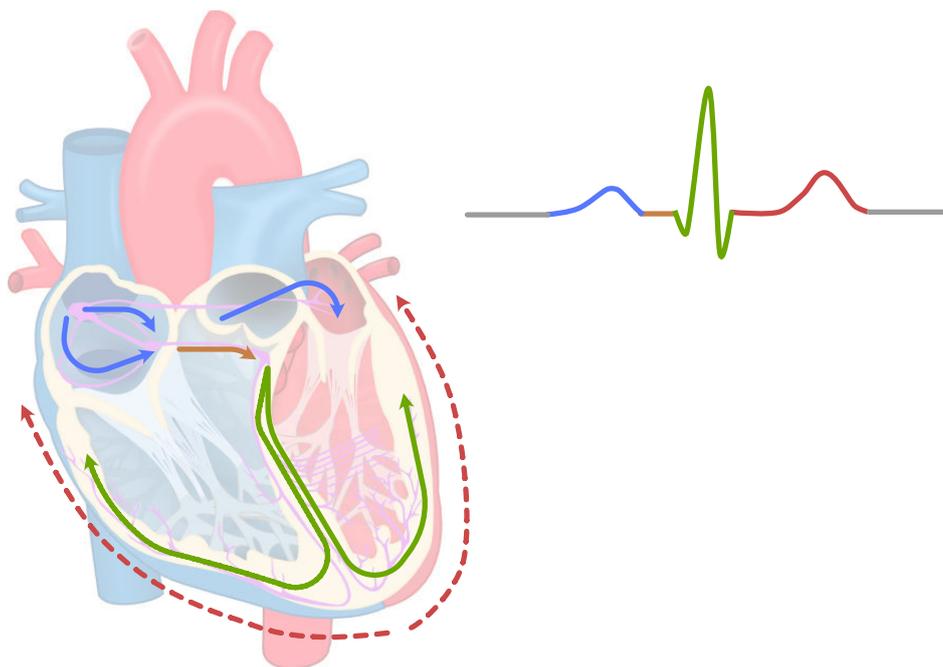


Figure 2.5: Anatomy of the heart, excitation conduction system, excitation spread and the representative parts in the ECG curve. The excitation starts in the atria (blue arrows) yielding in the P wave in the ECG. The PQ distance (brown) represents the conduction delay of the AV node. The QRS complex represents the depolarization of the ventricles (green). The repolarization of the ventricles is denoted by the T wave (red). Modified from [18] and from [19], both licensed under Creative Commons Attribution-Share Alike 3.0 Unported.

concentrations but also drug administration, second order diseases like hypertension, etc. [25] Chronic kidney disease (CKD) can be classified into five stages out of which four impair the patient's life. In stage I, a renal disease can be diagnosed, nevertheless, the renal function is not impaired. In stages II-IV the renal function is increasingly impaired until stage V representing the end-stage with a complete renal failure. A typical treatment can include drug therapy but also a renal replacement therapy as transplantation or dialysis [26, pp. 957-960]. The effects of CKD are various. In this work, the effect on the cardiovascular system is of special interest. Considering the relationship between cardiac excitation development and ionic currents described in Section 2.1.2, it gets apparent that changed ion, especially of K^+ and Ca^{2+} , concentrations directly affect the electrophysiological behavior of the heart cells. This results in a higher mortality through cardiovascular events in CKD patients than through the nephrological impairment itself [26, pp. 957-960].

In the end-stage of CKD, hemodialysis is a standard therapy for restoring the physiological ionic concentrations. Therefore, the principle of diffusion is used. The blood from the patient is driven into a dialysis (Figure 2.6) machine where the blood flow passes a semi-permeable membrane (Figure 2.7). On the other side of the membrane, a personalized dialysate consisting of demineralized water with added acetate or bicarbonate and electrolytes of

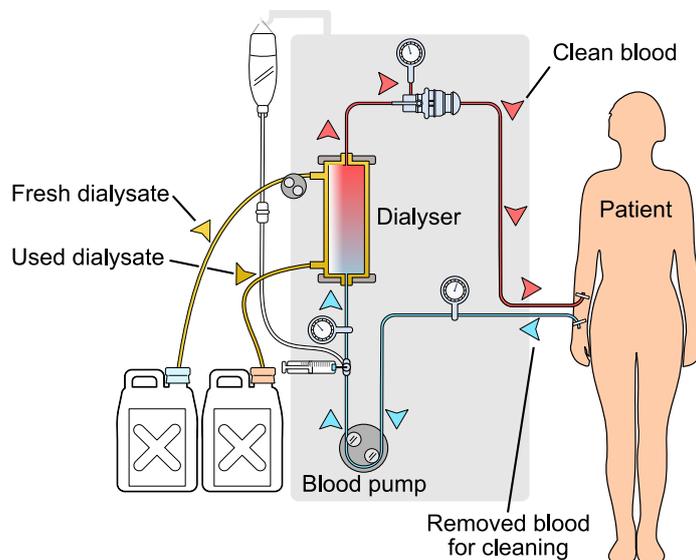


Figure 2.6: Concept of hemodialysis. Blood from the patient is pumped into the dialysis machine, cleaned and enters afterwards the body. For cleaning, the dialysate is used. Adapted from [27], licensed under Creative Commons Attribution-Share Alike 3.0 Unported.

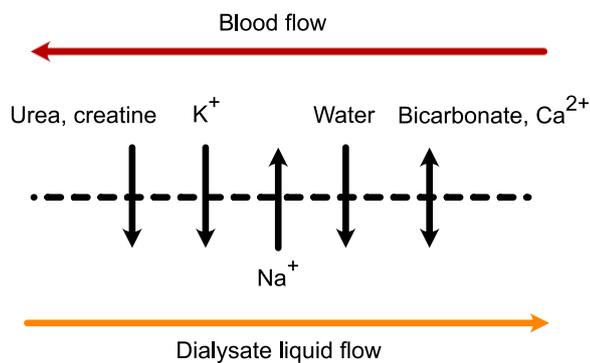


Figure 2.7: Concept of blood cleaning in the dialyser. Adapted from [28, p. 65].

different concentrations, adapted to the needs of the patient, passes the membrane. Usually, caused by the concentration gradient, Na^+ is transported into the blood, urea, creatine, water and K^+ is extracted from the blood [26, pp. 983-985].

Apart from hemodialysis, there are further concepts for dialysis, i.e. peritoneal dialysis or ultrafiltration. A detailed explanation can be found in medical textbooks [26, pp. 987-990].

2.4 Ectopic Sources and Ventricular Tachycardia

Ventricular tachycardia (heart rate > 160 bpm [26, p. 38]) can be caused (among other origins) by an ectopic source in the ventricular myocardium [23, p. 109]. In contrast to the physiological sinus rhythm having its origin in the sinus node, this ectopic source drives the excitation of the ventricles. In the case of a high excitation rate, this leads to ventricular tachycardia and consequently to death. Ventricular tachycardia can be classified as non-persistent if they last up to 30 s, and as persistent if they last longer than 30 s or if the tachycardia needs to be interrupted for hemodynamic reasons [26, p. 146].

Due to a changed excitation origin in the ventricles, the morphology of the ECG changes. P waves are not necessarily visible, QRS complexes and T waves change morphology dependent on the location of the ectopic center [23, pp. 108-109].

Dependent on the type of arrhythmia and the function of the left ventricle, the long-term success of a pharmacological treatment differs. If the arrhythmia cannot be compensated by anti-arrhythmical drugs, catheter ablation is the treatment of choice [26, p. 147]. During an electrophysiological treatment, the physician inserts a catheter into the venous system of the patient guiding it into the ventricles. Through mapping, i.e. the measurement of the potentials on the inner heart surface (endocardium) and additional stimulation maneuvers, the origin of the excitation can be determined. The tissue is then ablated (destroyed by either high or low temperature) to restore the physiological autonomous excitation [26, p. 155].

Technical Fundamentals

3.1 Electrophysiological Modeling

Electrophysiological modeling aims at describing mathematically the behavior of the heart cells and tissue. This can be achieved through detailed biophysical and more phenomenological modeling. In the following, selected approaches covering different degrees of granularity will be presented.

3.1.1 Cell Models

Cell models describe the electrophysiological behavior of cardiac cells, specifically the transmembrane voltage (TMV). Cell models usually build up on the differential equation [29, 30]

$$\frac{dV_m(t)}{dt} = -\frac{I_{ion}(t) + I_{stim}(t)}{C_m}, \quad (3.1)$$

where V_m is the TMV, C_m is the capacity of the cell membrane, I_{stim} is the stimulation current and I_{ion} is the ion current through the membrane. C_m is usually given as capacity per area ($[C] = F/m^2$), thus the currents are given as current densities ($[I] = A/m^2$). I_{ion} is defined as the sum of the currents of each ion channel through the cell membrane. The formulation of this variable is dependent on the cell model. The widely used ten Tusscher et al. [31, 32] models from 2004 and 2006 formulates this I_{ion} e.g. as

$$I_{ion} = I_{Na} + I_{K1} + I_{to} + I_{Kr} + I_{Ks} + I_{CaL} + I_{NaCa} + I_{NaK} + I_{pCa} + I_{pK} + I_{bCa} + I_{bNa}. \quad (3.2)$$

Here, I_{ion} consists of different sodium (Na^+), potassium (K^+), and calcium (Ca^{2+}) currents (determined by the respective indices) through the ion channels in the membrane, exchanger currents (I_{NaCa} , I_{NaK}), and pump currents (determined by an index p , e.g. I_{pK}). These currents are again formulated by individual expressions by the cell model. The formula for the L-type Ca^{2+} current in the ten Tusscher et al. model from 2006 is for example given by [32]

$$I_{CaL} = G_{CaL} \cdot d_V f_V f_{V2} f_{cass} \cdot 4 \cdot \frac{(V_m - 15)F^2}{RT} \frac{0.25[Ca^{2+}]_{SS} \cdot e^{2(V_m-15)F/RT} - [Ca^{2+}]_o}{e^{2(V_m-15)F/RT} - 1} \quad (3.3)$$

The L-type Ca^{2+} current is modeled depending on V_m , the constant maximum conductivity G_{CaL} , a voltage dependent activation gate d_V , two voltage dependent inactivation gates f_V and f_{V2} , an inactivation gate f_{cass} dependent on the subspace Ca^{2+} concentration $[Ca^{2+}]_{SS}$ and the constants R, F and T. The gating variables d and f represent the partial opening and closing of the channels, either dependent on the TMV or the intracellular Ca^{2+} concentration. The other currents for channels, exchangers, and pumps are modeled in a similar way yielding a set of coupled (differential) equations.

Dependent on the desired application, single currents in the I_{ion} formulation can be dropped, merged or even be refined. The latter was done in the work of Himeno et al. [33] for the formulation of the Ca^{2+} currents. Here, the authors introduced mainly two modeling steps:

1. Introduction of three intracellular subspaces with specific Ca^{2+} concentrations for each compartment (Figure 3.1).
2. Usage of Markov states (for further information see [34]) with three gating variables to describe the Ca^{2+} releasing units.

The first implementation was motivated by findings, e.g. of Acsai et al. [35], showing that near Ca^{2+} releasing sites, a Ca^{2+} concentration gradient can be observed in the cell. Himeno et al. discretized this cytosolic gradient by using three subspaces that are shown in Figure 3.1: bulk space (blk), an intermediate zone (iz), and a junctional space (jnc). Every subspace shows an individual Ca^{2+} concentration. As stated above and apart from the subspaces, the authors extended an existing Markov formulation by Hinch et al. [36] and Stern et al. [37] of the Ca^{2+} releasing units to be more realistic with respect to experimental results. This detailed Ca^{2+} modeling of the cell yields a more complex calculation of the current

$$I_{CaL} = \sum_x \sum_y I_{CaL_{y_x}} \text{ with } x \in \{jnc, iz, blk\} \text{ and } y \in \{Ca, Na, K\} \quad (3.4)$$

with

$$I_{CaL_{y_x}} = f_{CaL_x} \cdot P_{CaL_y} \cdot GHK_{y_x} \cdot pO_{LCC_x} \frac{1}{1 + \left(\frac{1.4}{6 \text{ mmol/l}}\right)^3} \quad (3.5)$$

Here, f_{CaL_x} , P_{CaL_y} are constants, GHK_{y_x} is the result of the used modified Goldman-Hodgkin-Katz equation, pO_{LCC_x} is the Markov chain probability for the opened L-type Ca^{2+} channel LCC_x gate. Comparing equation (3.3) to (3.4), it gets apparent that the computational demand increases compared to the widely used and above mentioned ten Tusscher model.

3.1.2 Excitation Propagation in the Tissue

Moving from microscopic single cell to macroscopic tissue simulations, it is not sufficient to consider only the behavior of each cell individually. Hence, it is crucial to model the excitation conduction in the connecting tissue, too. In the following, two approaches will be

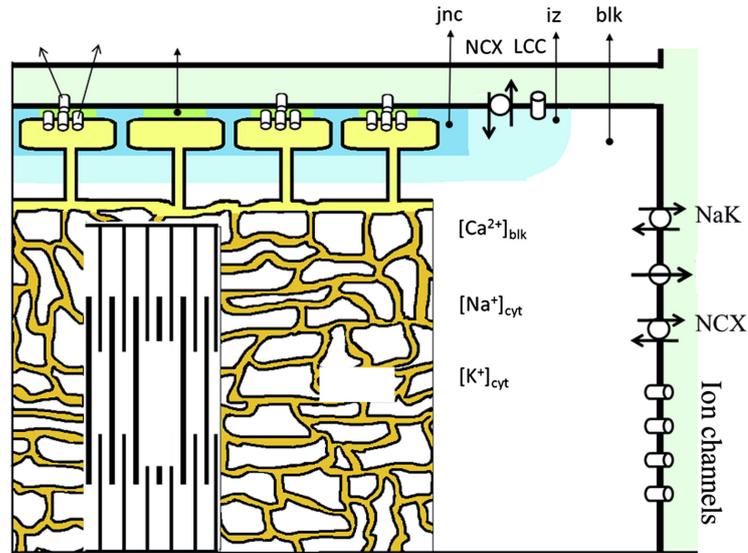


Figure 3.1: The concept of the Himeno et al. model [33]. The three subspaces are colored in light blue (iz), blue (jnc) and white (blk). L-type Ca^{2+} currents (LCC) are visualized with arrows for the different subspaces. NCX: sodium-calcium exchanger. NaK: sodium-potassium pump. The concentration index cyt denotes the cytosolic concentration. Adapted from Biophysical journal, vol. 109, no. 2, Himeno et al., A human ventricular myocyte model with a refined representation of excitation-contraction coupling, pp. 415-427, 2015, with permission from Elsevier. For the sake of clarity, several text blocks have been blanked in the figure. The figure has been cropped.

presented: a detailed biophysical model in Section 3.1.2.1 as well as a more phenomenological approach in Section 3.1.2.2.

3.1.2.1 Bidomain and Monodomain Model

The bidomain model was originally proposed by Tung [38] and is today the standard excitation propagation model for cardiac tissue. In this model, cardiac tissue is divided into two coupled domains: the intracellular and extracellular space, separated by the cell membrane. These two domains are characterized by anisotropic conductivity tensors $\sigma_{i/e}$ for intracellular (i) and extracellular (e) space, a current density per volume through the cell membrane i_m (i.e. from one domain to the other), current densities $\mathbf{j}_{i/e}$ and potentials $\Phi_{i/e}$ in the respective space. Poisson's equation delivers the following relationship

$$\nabla(\sigma_i \nabla \Phi_i) = i_m, \quad (3.6)$$

$$\nabla(\sigma_e \nabla \Phi_e) = -i_m. \quad (3.7)$$

With the two Poisson's equations, the definition of the TMV $V_m = \Phi_i - \Phi_e$ and the expression of i_m

$$i_m = \beta \left(C_m \frac{dV_m}{dt} + I_{ion} + I_{stim} \right), \quad (3.8)$$

where β is the cell surface to volume ratio, the two bidomain equations can be derived as

$$\nabla((\sigma_i + \sigma_e)\nabla\Phi_e) = -\nabla(\sigma_i\nabla V_m), \quad (3.9)$$

$$\nabla(\sigma_i\nabla V_m) + \nabla(\sigma_i\nabla\Phi_e) = \beta \left(C_m \frac{dV_m}{dt} + I_{ion} + I_{stim} \right). \quad (3.10)$$

The components of σ_i and σ_e are given in the parallel and perpendicular direction with respect to the direction of the myocardial fibers. If the ratio κ of the tensor components of σ_i and σ_e is equal, it can be written as $\sigma_i = \kappa\sigma_e$. Using this, a simplified expression of the bidomain equations, the monodomain equation

$$\frac{1}{1 + \kappa} \nabla(\sigma_i\nabla V_m) = \beta \left(C_m \frac{dV_m}{dt} + I_{ion} + I_{stim} \right) \quad (3.11)$$

is obtained. The calculation of the monodomain solution is computationally less expensive than of the bidomain solution.

3.1.2.2 Eikonal Equation and Fast Marching Algorithm

The excitation spread over the heart tissue can also be described phenomenologically [30, 39]. One well-established approach is solving the Eikonal equation

$$c\sqrt{\nabla t_a \mathbf{G} t_a} = 1 \quad (3.12)$$

with the speed $c(n)$ for each node n in an anatomical heart mesh, the node-wise activation time $t_a(n)$ and the tensor \mathbf{G} introducing anisotropy in the conduction properties similar to $\sigma_{i/e}$ in the bidomain model (Section 3.1.2.1). To solve the problem, i.e. calculating the activation times $t_a(n)$, the fast marching algorithm offers lower computational costs compared to Newton's method [40]. With the distribution of $t_a(n)$, the excitation spread over the heart tissue is determined.

3.1.3 Forward Problem of Electrocardiography

With the discrete formulation of the temporal and spatial TMV distribution in the heart tissue (e.g. from the discretized bidomain model), it is possible to extract the electrocardiogram (ECG). Therefore, the forward problem of electrocardiography has to be solved, which describes the mapping of sources from the heart (e.g. TMVs) to the body surface. The tissue between heart and body surface can be modeled as a volume conductor. Hence, Poisson's equation can be applied as before in the bidomain formulation [41] obtaining

$$\nabla(\sigma_i\nabla V_m) = -i_t. \quad (3.13)$$

The spatial gradient of the TMV induces a current density i_t into the tissue which itself serves as a source for the field potential Φ_e in the surrounding space

$$\nabla((\sigma_i + \sigma_e)\nabla\Phi_e) = i_t. \quad (3.14)$$

Assuming a torso with only passive resistive electrical properties, this delivers a linear and stationary way to calculate the potentials on the body surface with the TMV distribution in the heart [42]. The problem can be solved discretely using the finite element or boundary element method with two boundary conditions: a Dirichlet boundary condition caused by a reference electrode and a Neumann boundary condition using the boundary layer between thorax and air [43]. While the finite element method is computationally more expensive than the boundary element method, it offers the possibility to introduce a fixed distribution of different anisotropic conductivity regions, e.g. defined by different organs. The resulting surface ECG can be derived by evaluating the potentials Φ_e at certain predefined points (the desired electrode positions) on the body surface. In order to obtain the standard lead signals described in Section 2.2, these potentials have to be combined in a predefined way [44].

3.2 Regression and Classification Methods

In this section, selected methods for regression and classification will be described that were used in this work. They all belong to the class of supervised learning techniques, i.e. model fitting is based on given inputs (\mathbf{x}) and the corresponding known outputs (\mathbf{I}) which are often called labels. The pairs of known inputs and outputs is often referred to as training data.

3.2.1 Overfitting

Overfitting describes the phenomenon that a learning method performs well on the given inputs but fails to work well on new inputs [45]. The learning method only memorizes the training data and the generalization capability, i.e. performing also well on new inputs, is missing. This can be caused by an over-complex model or an unbalanced set. An example for overfitting is given in Figure 3.2. Six training data points (red circles) are used to approximate a sine wave (black dashed line). The 15th order polynomial fit (yellow line) is too complex and yields undesired results for points apart from the training points. The fit with the 3rd order polynomial (red line) better estimates the sine wave while the linear model is due to its low complexity not capable of showing the behavior of the sine wave.

3.2.2 Polynomial Regression

Polynomial regression forms one of the most intuitive regression techniques. Here, a model $y(\mathbf{x}, \mathbf{w})$ is parameterized using polynomial basis functions $\phi(\mathbf{x})$ [45, pp. 138-142]. This can be written as

$$y(\mathbf{x}, \mathbf{w}) = \sum_{j=0}^{M-1} w_j \phi_j(\mathbf{x}) = \mathbf{w}^T \phi(\mathbf{x}) . \quad (3.15)$$

Here, M is the total number of parameters in the model. To find the model parameters \mathbf{w} , the least squares method can be utilized. Hence, the known labels and inputs are used. Under the

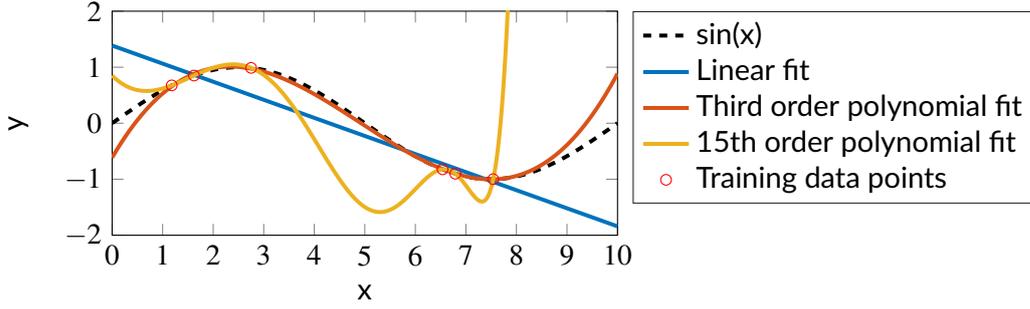


Figure 3.2: Regression of a sine wave (dashed) based on the given sparse training points. The used models were polynomials of first, third, and 15th order. This example is adapted from [45, p. 7].

assumption that the labels \mathbf{l} are realizations of $y(\mathbf{x}, \mathbf{w})$ superimposed with zero mean white Gaussian noise ε ,

$$\mathbf{l} = y(\mathbf{x}, \mathbf{w}) + \varepsilon, \quad (3.16)$$

a solution of the problem

$$\mathbf{w} = \underset{\mathbf{w}}{\operatorname{argmin}} \frac{1}{2} \|\mathbf{l} - \mathbf{w}^T \phi(\mathbf{x})\|_2^2 \quad (3.17)$$

is given by

$$\mathbf{w} = (\Phi^T \Phi)^{-1} \Phi^T \mathbf{l}. \quad (3.18)$$

The matrix Φ is called design matrix and has size $N \times M$, where N is the number of input-labels pairs available. The elements of the matrix are given as $\Phi_{ij} = \phi_j(x_i)$ where x_i is one specific input from the set of input-label pairs.

Depending on the pre-chosen order of the polynomial and the available data, overfitting might be a problem. Regularization can be used to reformulate the least squares problem (Equation (3.17)) and increase robustness. Adding a regularization term $\frac{\lambda}{2} \|\mathbf{w}\|_2^2$ to Equation (3.17) yields [45, pp.144-145]

$$\mathbf{w} = \underset{\mathbf{w}}{\operatorname{argmin}} \frac{1}{2} \|\mathbf{l} - \mathbf{w}^T \phi(\mathbf{x})\|_2^2 + \frac{\lambda}{2} \|\mathbf{w}\|_2^2. \quad (3.19)$$

The optimal solution in the least squares sense is given by

$$\mathbf{w} = (\lambda \mathbf{I} + \Phi^T \Phi)^{-1} \Phi^T \mathbf{l}. \quad (3.20)$$

This formulation penalizes both large deviations between model outputs $y(\mathbf{w}, \mathbf{x})$ and labels \mathbf{l} and large coefficient values \mathbf{w} . The regularization parameter λ controls the influence of the two L2-norms in Equation (3.19) on the solution. The incorporation of the squared sum of the coefficient values makes the method more robust against overfitting, since (especially with large polynomial model orders) the resulting high order model coefficients must stay as small as possible, consequently yielding smooth solutions [45, pp.144-145].

3.2.3 Support Vector Regression

Support vector regression (SVR) is another regression technique which is closely related to and inspired by the classifier support vector machine, a popular method for solving binary classification problems [46]. The basic idea behind SVR will be described in the following with a simplified example [46, 47]. The aim is to find a set of coefficients \mathbf{w} describing a linear function

$$f(\mathbf{w}, \mathbf{x}) = \mathbf{w}^T \mathbf{x} + b \quad (3.21)$$

for approximating N given label-input pairs (l_i, \mathbf{x}_i) . This is done not by minimizing the squared error as with polynomial regression but by minimizing the squared sum of the coefficients \mathbf{w} written as multiplication of the coefficient matrix

$$\begin{aligned} & \min_{\mathbf{w}} \frac{1}{2} \|\mathbf{w}\|_2^2, \\ & \text{subject to } \begin{cases} l_i - \mathbf{w}^T \mathbf{x}_i - b \leq \varepsilon & \text{for all } i = 1, \dots, N, \\ \mathbf{w}^T \mathbf{x}_i + b - l_i \leq \varepsilon & \text{for all } i = 1, \dots, N. \end{cases} \end{aligned} \quad (3.22)$$

The demand for small deviations between each label value l_i and the approximated value $f(\mathbf{w}, \mathbf{x}_i)$ is allowed to deliver an error up to a threshold ε . Nevertheless, the existence of a solution is not granted if $\varepsilon < \infty$ since some values $f(\mathbf{w}, \mathbf{x}_i)$ might be located outside of the ε band defined by the constraints in Equation (3.22). Therefore, the formulation in Equation (3.22) is extended by the individual errors (slack variables) ζ_i and ζ_i^* yielding

$$\begin{aligned} & \min_{\mathbf{w}, \zeta_i, \zeta_i^*} \frac{1}{2} \|\mathbf{w}\|_2^2 + C \sum_{i=1}^N (\zeta_i + \zeta_i^*), \\ & \text{subject to } \begin{cases} l_i - \mathbf{w}^T \mathbf{x}_i - b \leq \varepsilon + \zeta_i & \text{for all } i = 1, \dots, N, \\ \mathbf{w}^T \mathbf{x}_i + b - l_i \leq \varepsilon + \zeta_i^* & \text{for all } i = 1, \dots, N. \end{cases} \end{aligned} \quad (3.23)$$

The resulting situation is visualized in Figure 3.3. It gets apparent that only points outside the ε band (green dots) contribute to the cost function with their individual error ζ_i , the ones inside the gray area are neglected (black dots). The parameter C controls the trade-off between a small coefficient values \mathbf{w} (“flatness”) and the accepted degree of deviation from the threshold ε .

The optimization problem in Equation (3.23) can be converted into the dual formulation with the transformed variables α_i, α_i^* being advantageous for numerical computation and

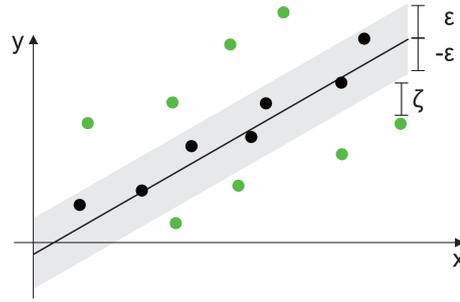


Figure 3.3: The concept of the SVR fit. The black line is the result of the fit. Only the green dots with the error ζ are considered during error calculation. All dots inside the ε region are not considered for the error calculation during the fitting procedure. The figure is adapted from [46].

obtaining a solution fast. The formulation is according to [46]

$$\begin{aligned} \max_{\alpha_i, \alpha_i^*} & -\frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N (\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*) k(\mathbf{x}_i, \mathbf{x}_j) & (3.24) \\ & -\varepsilon \sum_{i=1}^N (\alpha_i + \alpha_i^*) + \sum_{i=1}^N (\alpha_i - \alpha_i^*) l_i, \\ \text{subject to} & \sum_{i=1}^N (\alpha_i - \alpha_i^*) = 0 \quad \text{and} \quad \alpha_i, \alpha_i^* \in [0, C] \quad \text{for all } i = 1, \dots, N \end{aligned}$$

This can be solved by quadratic programming. In this dual formulation, a kernel transform $k(x_i, x_j)$ can be applied to make this method applicable to non-linear problems. When choosing the Gaussian kernel $k(x, \hat{x}) = \exp(-\frac{\|x - \hat{x}\|^2}{2\sigma^2})$, in total three hyperparameters ε , C , and σ (the standard deviation of the Gaussian kernel) have to be set. Furthermore, it should be emphasized that SVR is only capable of estimating a one-dimensional output as visible in Equation (3.21).

3.2.4 Artificial Neural Networks

Artificial neural networks (ANNs) is a further powerful method for regression and classification. In the following, basic principles will be explained.

3.2.4.1 Artificial Neuron

The artificial neuron is the element which ANNs consist of. Figure 3.4 shows the basic concept of a neuron. The inputs \mathbf{x} are fed into an input function $z(\mathbf{x}, \mathbf{w})$, characterized by the weights \mathbf{w} and a constant bias term w_0 . The output of z is passed to the activation function $f(z)$ which calculates the output of the neuron y . Input function and activation function can be chosen dependent on the network type [48, pp. 47-50].

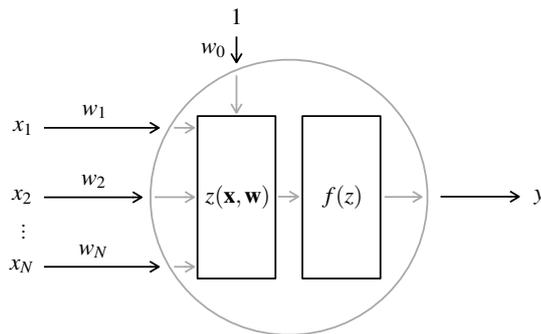


Figure 3.4: An artificial neuron with inputs x_j , weights w_j , an input function $z(\mathbf{x}, \mathbf{w})$, an activation function $f(z)$ and the output y .

3.2.4.2 Shallow Neural Networks

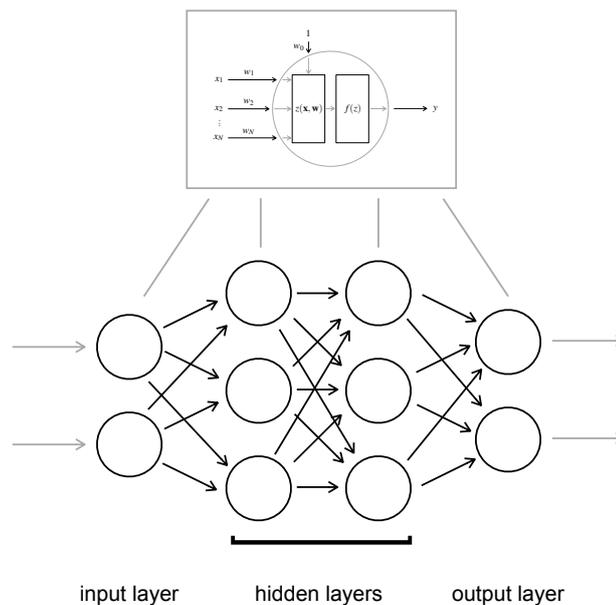


Figure 3.5: Typical ANN with an input layer, two hidden layers and one output layer. All connections between the layers are forwardly directed. Only preceding and succeeding layers are connected.

A shallow ANN consists of connected single neurons that are arranged in layers. Figure 3.5 shows an example for an ANN consisting of an input layer, two hidden layers and an output layer. Each layer consists of a predefined number of neurons. The number of input variables is equivalent to the number of neurons in the input layer, the number of neurons in the output layer is fixed by the number of output variables [48, pp. 122-129]. In addition to

the topology shown in Figure 3.5, it is possible to manipulate the interconnections between the neurons, e.g. introducing dropping connections, adding feedback loops, etc. [48, pp. 794-795] Nevertheless, it is important to prevent dead ends in the architecture.

One very common architecture is the multilayer perceptron (MLP) network. It is characterized by linear activation functions in input and output layers, differentiable non-linear activation functions in one or more hidden layers, and a high (commonly forwardly directed) connectivity [48, pp. 122-124]. In the following, a simple example will be utilized to explain the capability of ANNs to solve regression problems. Figure 3.6 shows an exemplary MLP with a two-dimensional input and a one-dimensional output. The number of hidden layers is one. The naming of the weights $w_{ij}^{[l]}$ in the figure follows a pattern: the upper index l is the number of the corresponding layer, the first lower index i determines the number of the neuron in preceding layer, the second index j the number of the neuron in the current layer. Input x and output y only depend on j and l . For this very simple example shown in Figure 3.6, it is possible to write down the functional connection between the input of the network ($x_1^{[1]}$ and $x_2^{[1]}$) and the output $y_1^{[3]}$. For the output of the neurons in the input layer, we can write:

$$y_1^{[1]} = f_1^{[1]}(z_1^{[1]}) = w_{11}^{[1]} \cdot x_1^{[1]} + w_{01}^{[1]} \quad (3.25)$$

$$y_2^{[1]} = f_2^{[1]}(z_2^{[1]}) = w_{22}^{[1]} \cdot x_2^{[1]} + w_{02}^{[1]} \quad (3.26)$$

$z_1^{[1]}$ is the linear input function of neuron 1 in layer 1, given by:

$$z_1^{[1]}(x_1^{[1]}, w_1^{[1]}) = w_1^{[1]} x_1^{[1]} + w_{01}^{[1]} \quad (3.27)$$

with $l = 1$ for the first layer. All other layers use the same input function. The activation function of layer $l = 1$ is linear: $f_j^{[l]} = z_j^{[l]}$. The activation function in layer 2 of neuron j , $f_j^{[2]}$ is usually defined as a sigmoid function of the form

$$f_j^{[2]} = \sigma(z_j^{[2]}) = \frac{1}{1 + e^{-z_j^{[2]}}} \quad \text{or alternatively to} \quad (3.28)$$

$$f_j^{[2]} = \sigma(z_j^{[2]}) = \frac{2}{1 + e^{-2z_j^{[2]}}} - 1. \quad (3.29)$$

The exact shape of the sigmoid can be chosen. With this (and following the red, respectively the blue path in Figure 3.6), the output of the hidden layer forms to

$$y_1^{[2]} = f_1^{[2]}(z_1^{[2]}) = \sigma(w_{11}^{[2]} \cdot y_1^{[1]} + w_{21}^{[2]} \cdot y_2^{[1]} + w_{01}^{[2]}), \quad (3.30)$$

$$y_2^{[2]} = f_2^{[2]}(z_2^{[2]}) = \sigma(w_{12}^{[2]} \cdot y_1^{[1]} + w_{22}^{[2]} \cdot y_2^{[1]} + w_{02}^{[2]}). \quad (3.31)$$

Finally, the output layer depends on the output of the hidden layer (grey arrows in Figure 3.6) and uses again a linear activation function

$$y_1^{[3]} = f_1^{[3]}(z_1^{[3]}) = w_{11}^{[3]} \cdot y_1^{[2]} + w_{21}^{[3]} \cdot y_2^{[2]} + w_{01}^{[3]}. \quad (3.32)$$

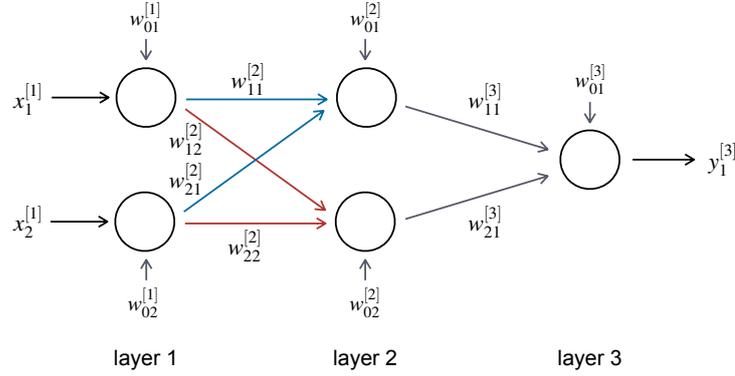


Figure 3.6: An MLP with two input neurons, one hidden layer consisting of two neurons and one output neuron.

The full expression for the output of the network

$$y_1^{[3]} = w_{11}^{[3]} \cdot \sigma(w_{11}^{[2]} \cdot w_{11}^{[1]} \cdot x_1^{[1]} + w_{01}^{[1]} + w_{21}^{[2]} \cdot w_{22}^{[1]} \cdot x_2^{[1]} + w_{02}^{[1]} + w_{01}^{[2]}) \quad (3.33)$$

$$+ w_{21}^{[3]} \cdot \sigma((w_{12}^{[2]} \cdot w_{11}^{[1]} \cdot x_1^{[1]} + w_{01}^{[1]} + w_{22}^{[2]} \cdot w_{22}^{[1]} \cdot x_2^{[1]} + w_{02}^{[1]} + w_{02}^{[2]}) + w_{01}^{[3]}) \quad (3.34)$$

is similar to the regression expression in Equation (3.15). The output of the ANN delivers a solution of a regression problem but utilizing sigmoid functions of a weighted linear combination of the inputs \mathbf{x} instead of polynomials $\phi(\mathbf{x})$ in Equation (3.15).

3.2.4.3 Optimization Methods

When applying neural networks as supervised learning strategy, the weights \mathbf{w} of the network have to be determined in such a way that the output of the network $\mathbf{y} = f_{net}(\mathbf{w}, \mathbf{x})$ fits given labels \mathbf{l} with respect to given inputs \mathbf{x} . This is called training in machine learning terminology. The estimation error is determined by the so-called loss function $L(\mathbf{y}, \mathbf{l})$ which should be minimal [49, pp. 82–83]. The optimization problem which is solved during training states as

$$\mathbf{w} = \underset{\mathbf{w}}{\operatorname{argmin}} L(f_{net}(\mathbf{w}, \mathbf{x}), \mathbf{l}) . \quad (3.35)$$

The loss function $L(\mathbf{y}, \mathbf{l})$ is chosen dependent on the specific task and can exemplary be chosen to

$$L(\mathbf{y}, \mathbf{l}) = \|\mathbf{y} - \mathbf{l}\|_2^2 \quad (3.36)$$

for a regression task. In the following, three methods will be described to obtain the optimal weights \mathbf{w} (obtaining a minimal loss) during training.

Gradient Descent To obtain a locally optimal solution, gradient descent can be applied. The idea behind this algorithm is to use the gradient of the function $L(f_{net}(\mathbf{w}, \mathbf{x}), \mathbf{l})$. The

function, respectively the gradient, describes the behavior of the network and is used to change \mathbf{w} iteratively in such a way that the loss function gets minimal (Equation (3.35)). Starting with \mathbf{w}_j , the iterative process of finding the next set of weights \mathbf{w}_{j+1} can be described by the following expression [49, pp. 82–86] [50, pp. 35–36]

$$\mathbf{w}_{j+1} = \mathbf{w}_j - \eta \nabla_{\mathbf{w}} L(F(\mathbf{w}_j, \mathbf{x}), \mathbf{l}) . \quad (3.37)$$

It gets apparent, that an initial weight vector \mathbf{w}_0 needs to be preset, e.g. by random initialization. Furthermore, the step size (or learning rate) η needs to be given.

Stochastic Gradient Descent When training neural networks consisting of millions of weights, the gradient descent method might run into problems. Large datasets are required and due to memory constraints, it might be hard to calculate the optimal solution. However, it can be proven that randomly splitting the whole dataset into subsets, and calculating the gradient on just one subset solves the problem as well [49, pp. 271–292]. Including these steps, Equation (3.37) changes to

$$\begin{aligned} \mathbf{w}_{j+1} &= \mathbf{w}_j - \eta \nabla_{\mathbf{w}} L(F(\mathbf{w}_j, \mathbf{x}), \mathbf{l}) = \mathbf{w}_j - \eta \nabla_{\mathbf{w}} \sum_{i=1}^N (y_i - l_i)^2 = \mathbf{w}_j - \eta \sum_{i=1}^N \nabla_{\mathbf{w}} (y_i - l_i)^2 \quad (3.38) \\ &\approx \mathbf{w}_j - \eta \sum_{i=1}^M \nabla_{\mathbf{w}} (y_i - l_i)^2 , \end{aligned}$$

where N is the number of label-input pairs, $M < N$ is the number of samples in the subset. Stochastic gradient descent uses the fact that the gradient can be approximated by calculating the gradient with only a selected part of the given data which however have to be chosen randomly in each iteration. This is then used to update all weights. The subset of label-input values which is used for the calculation of the gradient is called minibatch. When applying stochastic gradient descent in practice with ANNs, it is important to decrease the learning rate η over the iterations as the random minibatch selection introduces a source of noise which does not vanish when arriving at the minimum [49, pp. 151–152, 271–292].

ADAM The ADAM algorithm was proposed by Kingma and Ba in 2014 [51]. Their idea was to calculate the parameter update dependent on the calculated gradient $g_{j+1} = \nabla_{\mathbf{w}} L(F(\mathbf{w}_j, \mathbf{x}), \mathbf{l})$, the element-wise squared gradient g_{j+1}^2 and preceding values for those. The ADAM update expression is defined as

$$\mathbf{w}_{j+1} = \mathbf{w}_j - \eta \frac{\hat{m}_{j+1}}{\sqrt{\hat{v}_{j+1} + 1 + \gamma}} . \quad (3.39)$$

Here, γ is a small constant that prevents the denominator to get zero. The variables \hat{m} and \hat{v} are defined as

$$\hat{m}_{j+1} = \frac{m_{j+1}}{1 - \beta_1^{j+1}} , \quad (3.40)$$

$$\hat{v}_{j+1} = \frac{v_{j+1}}{1 - \beta_2^{j+1}} , \quad (3.41)$$

with

$$m_{j+1} = \beta_1 \cdot m_j + (1 - \beta_1) \cdot g_{j+1} , \quad (3.42)$$

$$v_{j+1} = \beta_2 \cdot v_j + (1 - \beta_2) \cdot g_{j+1}^2 . \quad (3.43)$$

The parameters β_1 and β_2 have to be pre-chosen, m_0 and v_0 have to be initialized. The novelty of this method was the combination of the minibatch concept from stochastic gradient descent and the inclusion of preceding gradients in the weight update calculation.

3.2.4.4 Back-propagation Algorithm

During the training of ANNs, the so-called back-propagation algorithm is applied [48, pp. 129-131]. The idea is to use the chain rule to find an expression for the gradient $\frac{\partial L}{\partial w_i^k}$, where w_i^k is a weight in a layer k , being dependent on only the weights and outputs of succeeding layers. Considering this, the optimization problem originally stated in Equation (3.35) can be solved by a layer-wise optimization (computation of the gradients). This finding is exploited in the back-propagation algorithm which works as follows [48, pp. 139-141]: First, all weights are initialized and all needed outputs of the single layers are computed. Afterwards, the actual back-propagation is performed: with the error of the output layer calculated from the known targets, the errors in all preceding layers are calculated successively gaining the updated weights. With the new set of weights, all the variables are calculated again and the procedure is repeated until a stopping criterion is hit.

The calculation of the errors of a single neuron in one layer is independent of the other neurons in that layer permitting parallel computation of those.

3.2.4.5 Regularization

A common problem in machine learning is overfitting. To avoid this in ANNs, regularization can be used. Two common methods are presented here:

Bayesian Neural Networks The idea behind Bayesian neural networks is to add a regularization term E_B to the cost function (Equation (3.35)) in the optimization yielding

$$L_{Reg} = \frac{\alpha}{2} L(\mathbf{y}, \mathbf{l}) + \frac{\beta}{2} \|\mathbf{w}\|_2^2 . \quad (3.44)$$

α and β are weightings optimized during the learning process and depend on the errors and weightings, respectively. The purpose of this extension is to keep the norm of the weights small, favoring in a smoother and simpler solution similar to the regularization approach described in Section 3.2.2. Hence, overfitting can be reduced [45, pp. 277-284][52].

Early Stopping With early stopping it is tried to detect overfitting before it happens. Therefore, a part of the training dataset is used for the validation (validation dataset) of the loss during training. When the loss on this subset which is not be used for the calculation of

the optimal \mathbf{w} increases for a predefined number of training iterations, training is stopped. This aims at decreasing the error with an unseen test dataset [45, pp. 259-261].

3.2.5 Convolutional Neural Networks

When applying the regression and classification techniques described until now, it is not common to use, e.g. the samples of an ECG signal directly as input. More, features describing a certain change that is expected to be visible, are calculated in advance and these are used as inputs for the learning method. Shallow neural networks were shown to solve a large variety of existing problems. Nevertheless, they were found to deliver unsatisfying results in pattern recognition. The algorithms are not able to generalize well, i.e. performing well on data different from training samples. To overcome this in an efficient way, deep learning was designed [49, p. 155]. The expression is not clearly defined but covers ANNs consisting of many layers making them “deep”.

Convolutional neural networks (CNNs) form a class of neural networks that are usually designed as deep neural networks. The main idea behind CNNs is to use convolutional layers for an automatic feature extraction. These convolutional layers apply different convolutional kernels on a given input which can be a two-dimensional signal for example. The definition of the convolutional kernels becomes part of the optimization process. The CNN learns how to perform these convolutions in such a way that the problem from Equation (3.35) is solved. Figure 3.7 visualizes the concept of a CNN. The input is a multi-lead ECG signal which is visualized as an image. Here, lead signals are arranged horizontally, vertically time is visible. The amplitude is gray value coded. The upper part of the image showing many white and black spots is the part of the QRS complex. A set of convolutional operations (with different convolutional kernels) is applied to this two-dimensional input. The results from the convolutions forms a set of outputs, called feature maps. After a downsampling (usually referred to as pooling), the convolutional operation is repeated as the downsampling is. When the information is reduced to a set of images of size 1×1 , i.e. single values, these are fed into a fully connected layer (similar to the architecture shown in Section 3.2.4.2) which calculates the output of the network. The weights of the final fully connected layer, as well as all the convolutional kernel coefficients are the optimized parameters during training [49, pp. 339-341].

As with shallow neural networks, deep neural networks utilize the back-propagation algorithm during training making them highly parallelizable. Nevertheless, the high complexity of these networks with regard to their number of parameter values (weights) enforces the application of regularization (e.g. early stopping) and special optimization techniques as stochastic gradient descent or ADAM (Section 3.2.4.3). More information is given in [49, pp. 274-329].

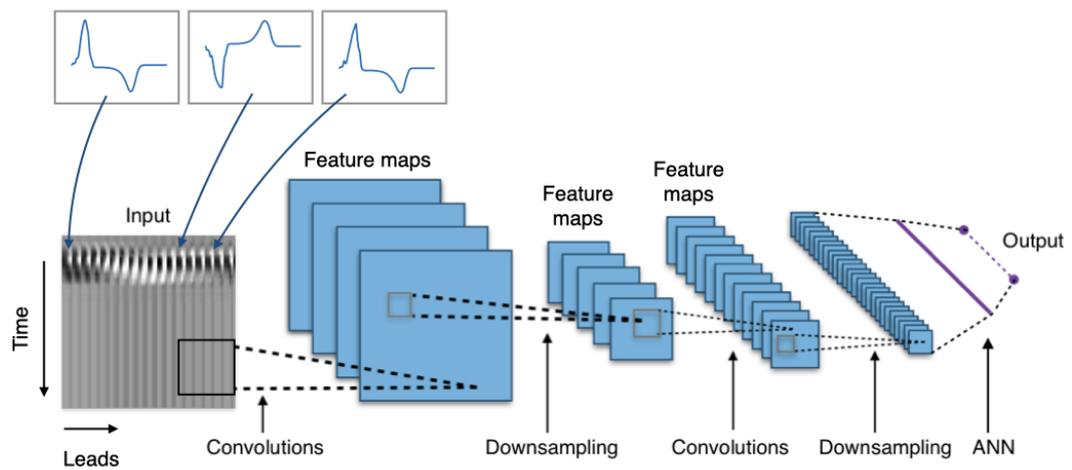


Figure 3.7: Concept of a CNN. The two dimensional input image consists of different ECG lead signals concatenated vertically. The input is passed through several convolutional and downsampling steps obtaining different feature maps. Finally, several 1x1 images are fed into a fully connected layer yielding the output. Adapted from [53], licensed under the Creative Commons Attribution-Share Alike 4.0 International license.

PART II

IONIC CONCENTRATION ESTIMATION WITH THE ECG

Introduction to the Topic

The disturbance of the blood ion concentrations is a relevant clinical issue. In a study comprising 364,955 patients accessing healthcare in Stockholm, Sweden, 13.6% of the patients were found to suffer from hypokalemia ($[K^+] < 3.5$ mmol/l), 2.5% showed moderate or severe hyperkalemia ($[K^+] > 5.5$ mmol/l) [54]. As with potassium, the impairment of other electrolytes can only be diagnosed with blood samples. A non-invasive monitoring technique at the point-of-care is desirable allowing an early diagnosis, a rapid therapy start and consequently an optimized patient outcome [54]. Nevertheless, apart from the emergency setting, patients with chronic diseases causing the impairment of the electrolyte concentrations may also benefit from a rapid diagnosis. For example, chronic kidney disease (CKD) patients treated with haemodialysis undergo severe blood electrolyte changes during the dialytic and inter-dialytic phases. These go hand-in-hand with a 14-fold increased risk of dying from sudden cardiac death (SCD) compared to patients without renal impairment but with pre-diagnosed cardiovascular diseases [3]. The reasons for this are various and not entirely understood, but electrolyte changes are assumed to be one important factor [8–11]. Complications can be observed not only during the haemodialysis sessions but also during the inter-dialytic phase [5–7]. Especially in this latter phase, the patient is usually not hospitalized making concentration measurements impossible and preventing a final evaluation of the influence of concentration fluctuations and cardiovascular events. The electrocardiogram (ECG) as a non-invasive monitoring device which is already used for home monitoring in other diagnostic domains, could be utilized for a continuous measurement of the patient's ionic concentration values. This idea is obvious since blood calcium concentration ($[Ca^{2+}]_b$) and blood potassium concentration ($[K^+]_b$) are important players in the excitation mechanisms of the heart which can be captured by the ECG. Furthermore, such a technique could support the research on the underlying mechanism being responsible for the increased cardiovascular disease (CVD) incidence in CKD patients [55]. There have been approaches for ECG-based $[K^+]_b$ estimation [56–59] (Table 4.1). Apart from the lack of studies estimating $[Ca^{2+}]_b$, there are still open problems with the $[K^+]_b$ estimation as the errors in Table 4.1 suggest. The aim of this project was: analyzing the shortcomings of existing approaches, revealing hidden influences leading to the errors and proposing new workflows for the concentration estimation. Furthermore, a possible $[Ca^{2+}]_b$ estimation was

Table 4.1: Methods for $[K^+]_b$ estimation from literature. CNN: Convolutional neural network, $T_{s/a}$: T downslope divided by T amplitude, $T_{s/\sqrt{a}}$: T downslope divided by the square root of T amplitude, Result: mean \pm standard deviation of unsigned errors in mmol/l, Dataset: distribution of $[K^+]_b$ in the analyzed dataset, n/a: not available/given. The table is taken from [60].

Work	Lead(s)	Features	Result (mmol/l)	Dataset (mmol/l)
Corsi et al. [56]	PCA	$T_{s/A}$	0.46 ± 0.39	n/a
Attia et al. [57] personalized	V3-V5	$T_{s/\sqrt{A}}$	0.36 ± 0.34	4.2 ± 0.95
Attia et al. [57] global	V3-V5	$T_{s/\sqrt{A}}$	0.50 ± 0.42	3.9 ± 0.8
Yasin et al. [58] personalized	I	$T_{s/\sqrt{A}}$	0.38 ± 0.32	4.3 ± 0.8
Lin et al. [59]	12-lead	CNN	$0.53\pm n/a$	n/a

investigated. All investigations were based on simulations and clinical data. The latter were provided by courtesy by Stefano Severi and Cristiana Corsi from the University of Bologna and parts of the dataset was used by them in [56].

Simulation Studies

5.1 Himeno et al. Whole Heart Simulations

As stated in Chapter 4, the existing algorithms for blood potassium concentration ($[K^+]_b$) estimation still show mean absolute errors in the range of 0.5 mmol/l. The exact reasons for these can be various. Computational modeling offers the possibility to minimize confounding (partly random) factors in the signal acquisition to get a clear view on the effects caused by $[K^+]_b$ changes. This enables an analysis of the workflow and the proposal of new optimized methods. Furthermore, it enables the generation of data that are not available yet. As measurements from patients with large blood calcium concentration ($[Ca^{2+}]_b$) variations are rare, this fact allows for the development of a $[Ca^{2+}]_b$ estimation method based on the simulated signals.

In the following, the generation of a simulated electrocardiogram (ECG) dataset with different underlying $[K^+]_b$ and $[Ca^{2+}]_b$ is described. These data were subsequently utilized to analyze, optimize and continue to develop a workflow for concentration estimation. *This work was done in collaboration with Axel Loewe and María Hernández Mesa. Results were published in [61–63].*

The widely used cellular models proposed by ten Tusscher et al. [31, 32] and O’Hara et al. [64] fail to reproduce a correct relation between action potential (AP) duration and extracellular calcium concentration ($[Ca^{2+}]_o$) [65–68]. Consequently, they could not serve as basis for realistic ECG simulations with changing $[Ca^{2+}]_o$. Instead of adjusting the proposed models, the Himeno et al. ventricular cell model was used. This is described as a single cell model only [33], however, as described in Section 3.1.1, it offers a more realistic description of the calcium (Ca^{2+}) handling in the cell, e.g. by the introduction of compartments modeling the spatial Ca^{2+} concentration gradients known to exist.

To be able to perform a full ventricular simulation with a following ECG extraction, mainly two things have to be done: first, a formulation of the transmural heterogeneity, i.e. adding epicardial and midmyocardial cell type formulations to the model, needs to be found as stated in [69]. Second, a spatial gradient of the I_{K_s} channel conductivity needs to be incorporated

into the model to achieve a realistic T wave [69, 70]. In the following, the two steps will be presented.

Transmural Heterogeneity The Himeno et al. model offers a formulation for the endocardial cell types. Nevertheless, for transmural heterogeneity, two other cell types, epicardial and midmyocardial cells, are usually described. The introduction of the missing two cell types was done according to the formulation of O’Hara et al. [64]. Table 5.1 lists the parameters that were changed to introduce transmural heterogeneity, for both the O’Hara et al. model and the adjusted version for the Himeno et al. model [61]. Variable names differed in both articles so the corresponding abbreviations are given together with the ratios of epicardial to endocardial and midmyocardial to endocardial values.

Table 5.1: Ratios of epicardial (epi) to endocardial (endo), and midmyocardial (m) to endocardial (endo) parameters describing the transmural heterogeneities in the O’Hara et al. model [64] and in the adjusted formulation of the Himeno et al. model.

Parameter names		epi/endo		m/endo	
O’Hara et al.	Himeno et al.	O’Hara et al.	Himeno et al.	O’Hara et al.	Himeno et al.
G_{NaL}	P_{Na}	0.6	0.6	1	1
G_{to}	G_{Kto}	4	4	4	4
$P_{Ca}, P_{CaNa}, P_{CaK}$	P_{CaL_Ca}	1.2	1.2	2.5	2
G_{Kr}	G_{Kr}	1.3	1.3	0.8	1
G_{Ks}	P_{Ks_K}	1.4	1.4	1	1
G_{K1}	G_{K1}	1.2	1.2	1.3	1.3
$G_{NaCa,i}, G_{NaCa,ss}$	Amp_{NCX}	1.1	1.1	1.4	1.4
G_{NaK}	Amp_{NaK}	0.9	0.9	0.7	1.5
G_{Kb}	P_{bNSC_K}	0.6	0.6	1	1
$J_{rel,NP,\infty}, J_{rel,CaMK,\infty}$	P_{RyR}	1	1	1.7	1.4
$J_{up,NP}, J_{up,CaMK}$	Amp_{SERCA}	1.3	1.3	1	1

$$\delta_{epi} = 1.0 - \frac{0.95}{1.0 + \exp\left(\frac{V+70.0}{5.0}\right)} \quad (5.1)$$

$$\tau_{i,epi,fast} = \tau_{i,fast} \cdot \delta_{epi} \quad (5.2)$$

$$\tau_{i,epi,slow} = \tau_{i,slow} \cdot \delta_{epi} \quad (5.3)$$

The ratios between epicardial cells and endocardial cells could be applied to the Himeno et al. as it was proposed by O’Hara et al. Nevertheless, for the ratio of midmyocardial to endocardial, adjustments had to be made. These comprised the Himeno et al. parameters P_{CaL_Ca} , P_{RyR} , Amp_{NaK} and G_{Kr} yielding to a stable AP with a pronounced overshoot, an adequate repolarization and no early or delayed afterdepolarizations. These adjustments stayed in the range of experimental data. Delayed afterdepolarizations, i.e. spontaneous depolarization during AP phase two or three (Figure 2.3), could only be avoided by the

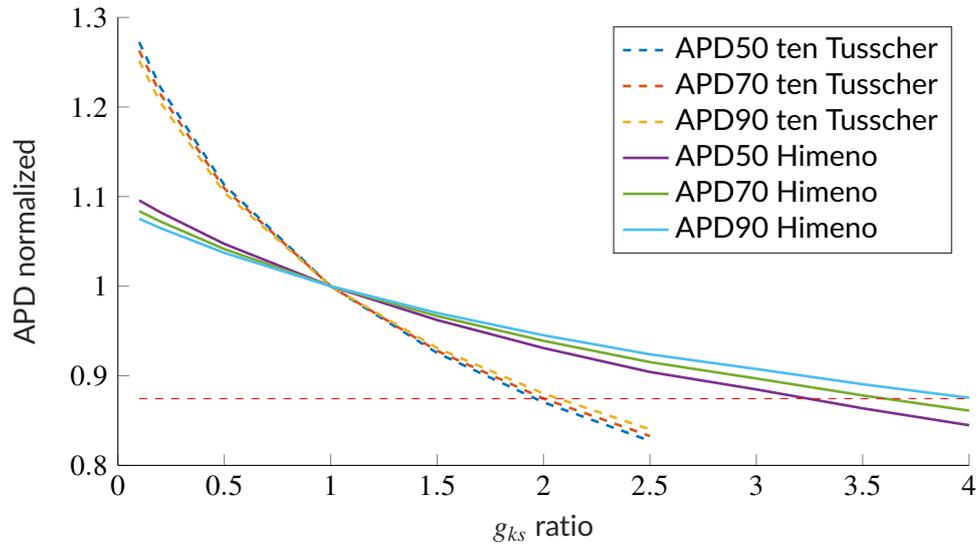


Figure 5.1: Relationship between AP duration and the g_{Ks} ratio between base and apex. The AP duration is normalized to the AP value for a g_{Ks} ratio of 1.

adjustment of Amp_{NaK} . This seemed to be legitimate as the restitution behavior of the model was still according to findings from [71].

Spatial I_{Ks} Gradient As proposed by Keller et al. [69], the experimentally found spatial variation of the AP duration can be modeled by a spatial gradient of the conductivity g_{Ks} determining the current I_{Ks} . Seemann et al. applied the ten Tusscher et al. model [32] with a gradient of twice the g_{Ks} conductivity of the base of the heart compared to the conductivity at the apex with a linear course between those two regions [70]. This gradient was scaled appropriately for the Himeno et al. model by demanding the AP durations at base and apex to be equivalent to those visible with the ten Tusscher et al. model. Figure 5.1 visualizes the relationship of different AP durations, the models applied and the conductivity factor of g_{Ks} with respect to the conductivity at the base of the heart. The factor for the Himeno et al. model was 3.5 achieving equivalent AP durations as with the ten Tusscher et al. model used in [70].

With these adjustments, it was possible to perform monodomain simulations with the simulation framework acCELLerate [72]. The anatomical models from [70] were used. The resulting simulated transmembrane voltages on the heart were forward calculated with an inhomogeneous torso model including separate conductivities for heart, lung and liver using the finite element method. Last, the standard 12-lead ECG was extracted from the voltage distribution on the torso surface. 81 simulations were conducted varying $[\text{K}^+]_b$ in the range of 3 mmol/l and 7.8 mmol/l and $[\text{Ca}^{2+}]_b$ between 0.8 mmol/l and 3 mmol/l. The exact list of simulations can be found in Table A.1.

5.2 Optimization of Pre-processing Steps

With the simulations from the section before, it was possible to answer the following three questions:

1. Can existing feature extraction methods be improved?
2. How do noisy signals need to be filtered to reduce noise efficiently but not distort the feature values?
3. What is the optimal choice of a lead reduction technique?

The answers helped to optimize the workflow used for concentration estimation. The simulated ECGs additionally offered the possibility to quantify improvements and problems connected to these questions as there is always the ground truth available.

5.2.1 Improvement of Feature Extraction

5.2.1.1 Motivation

Feature extraction is a crucial step in the analysis of ECG signals. Although it is used in several studies, e.g. presenting arrhythmia classification [73], the actual evaluation of feature extraction methods is very often missing. This fact seems plausible as it is very hard to generate a ground truth dataset including values for features like slopes from the T wave. However, simulated signals can help to circumvent this problem with regard to the evaluation of the robustness. If the feature extraction algorithm works robustly, it has to show the following two characteristics:

1. The algorithm has to deliver a similar or the same result for noise-free and noisy signals.
2. Visually visible trends (e.g. an amplitude decrease caused by a decreasing $[K^+]_b$) have to be visible in the feature progression.

As the extracted ECGs from the simulations are noise-free and noise can easily be induced artificially, the first property can be checked easily. The second requirement can be checked by the analysis of functions whose parameters can induce feature changes directly, e.g. the standard deviation of a Gaussian function.

5.2.1.2 Methods

81 simulated ECGs with different $[K^+]_b$ and $[Ca^{2+}]_b$ were fed into the feature extraction methods proposed in [74, 75]. These algorithms were designed to compute 14 features on simulated signals. The features were:

1. first statistical moment of the T wave distribution (T center)

2. second statistical moment of the T wave distribution (T variance)
3. third statistical moment of the T wave distribution (T skewness)
4. fourth statistical moment of the T wave distribution (T kurtosis)
5. distance between R and T wave peak (RT distance)
6. peakedness of the T wave (T peakedness)
7. T wave amplitude (T amplitude)
8. slope of the ascending part of the T wave (T upslope)
9. slope of the descending part of the T wave (T downslope)
10. ratio of second half T wave energy and whole T wave energy (T ratio 2nd half)
11. ratio of first half T wave energy and whole T wave energy (T ratio 1st half)
12. R peak amplitude (R amplitude)
13. R peak energy (R energy)
14. ratio R peak energy and R peak amplitude (R ratio en. to amp.)

For several features, different ways to calculate the value can be imagined, e.g. for the slopes of the T wave, it remains unclear, how to exactly calculate one specific value as the slope is still a temporal function. Moreover, it is not clear how these feature extraction algorithms perform on noisy signals. To estimate the performance of this initial implementation, white Gaussian noise was added to the simulated ECGs obtaining a signal to noise ratio (SNR) of 20 dB. The resulting signals were then low-pass filtered with a fourth order zero phase Butterworth filter (cut-off frequency 80 Hz) to generate signals that are comparable to the clinical setting. Features were calculated for these signals, as well as for the noise-free signals. The relative error was taken as the measure to evaluate the robustness of the feature extraction algorithms. After the analysis of the initial implementation, selected further calculation methods (if available) for the single features were evaluated. The most robust approach was selected for the further analysis.

5.2.1.3 Results

The performance of the initial implementation of the feature extraction is shown in Figure 5.2. It gets apparent that temporal and amplitude features perform quite robust. Nevertheless, even for a high but realistic SNR of 20 dB [76–78], the interquartile range of the relative error of the fourth statistical moment, the peakedness, T ratio 2nd half, RT distance, and the slopes of the T wave exceed 20%. This is problematic as the downslope is one of the most frequently used features for $[K^+]_b$ estimation.

One well-established way to increase robustness is introducing a-priori knowledge into the algorithm. Hence, T downslopes and upslopes were approximated with a polynomial fit of fourth order before calculating the slope. Figure 5.3 shows the fitted functions for the up- and downslope of the T wave. It gets apparent that the slope calculation from the smooth model fit can be more robust. In fact if this implementation was compared to the initial calculation of the slopes, the robustness of the slope calculation increased (Figure 5.5(a) and Figure 5.5(c)).

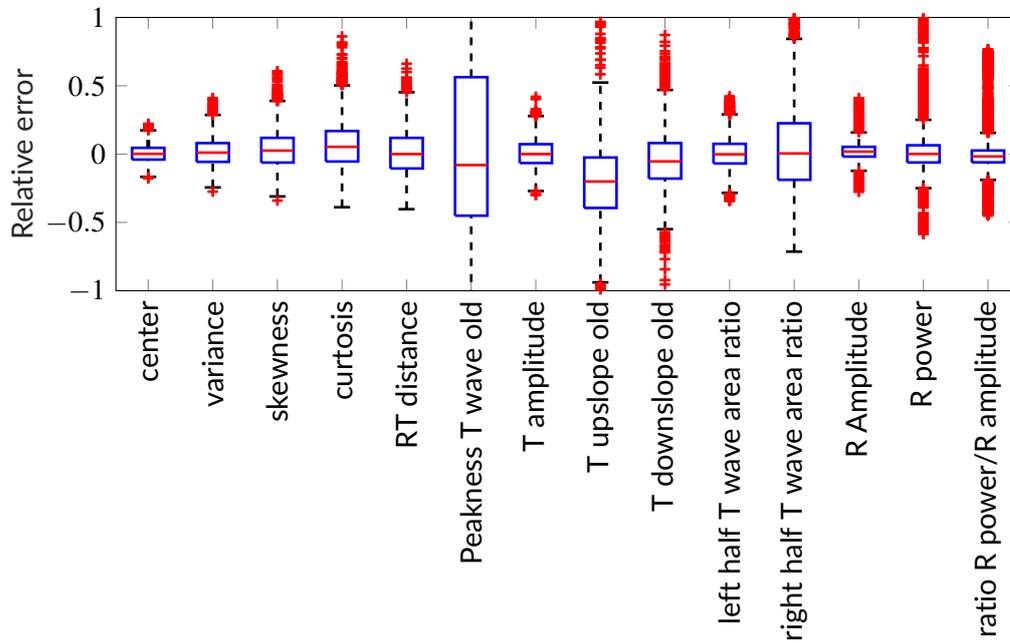


Figure 5.2: Relative errors with regard to the noise-free signals of the single features. The less robust features regarding the interquartile range are the third and fourth order statistical moments, T peakedness, T ratio 2nd half, RT distance, and the slopes of the T wave.

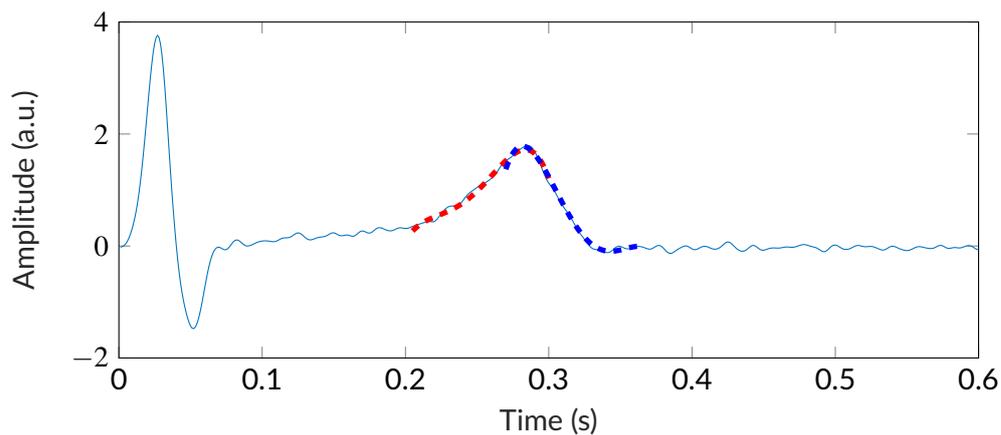


Figure 5.3: Noisy simulated ECG signal with the polynomial fits for both upslope calculation (red) and downslope calculation (blue).

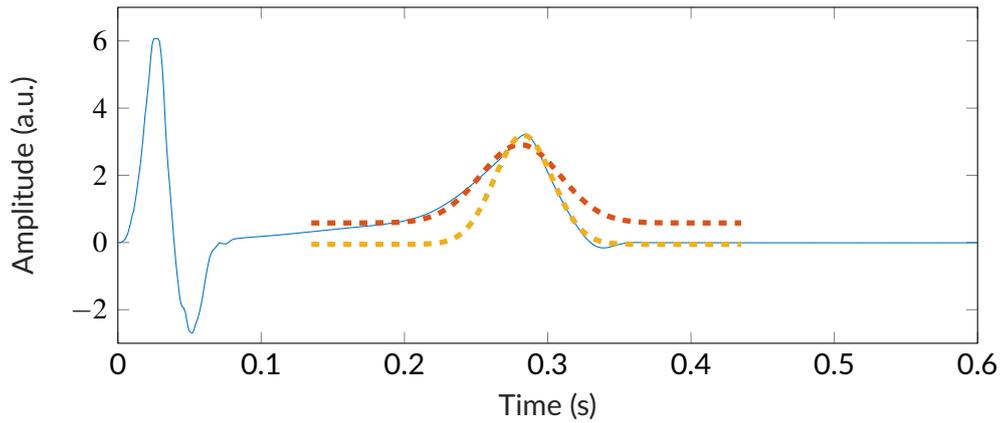


Figure 5.4: Simulated ECG signal with two Gaussian fits for the ascending part (red dashed line) and for the descending part (yellow dashed line). The two Gaussians have different standard deviations and offsets that are used for the calculation of the peakedness and the ST change.

Apart from the slopes, two further features were calculated based on model parameters: an improved version of T peakedness and ST elevation. Therefore, two Gaussian functions of the form $f(x, a, \mu, \sigma, b) = ae^{-\frac{(x-\mu)^2}{2\sigma^2}} + b$ were fitted to the ascending and descending part of the T wave (Figure 5.4). The peakedness (Figure 5.5(b) which is now the inverted sum of the two standard deviations σ of the Gaussians (the lower the two standard deviations are, the narrower the T wave is), and the ST change (elevation or depression) which is the difference of the fitted offset parameters b , were determined with the model parameters. The determination of the slopes with the Gaussian model fit delivered better results than the original approach but less robust results than the selected fourth order polynomial fit approach.

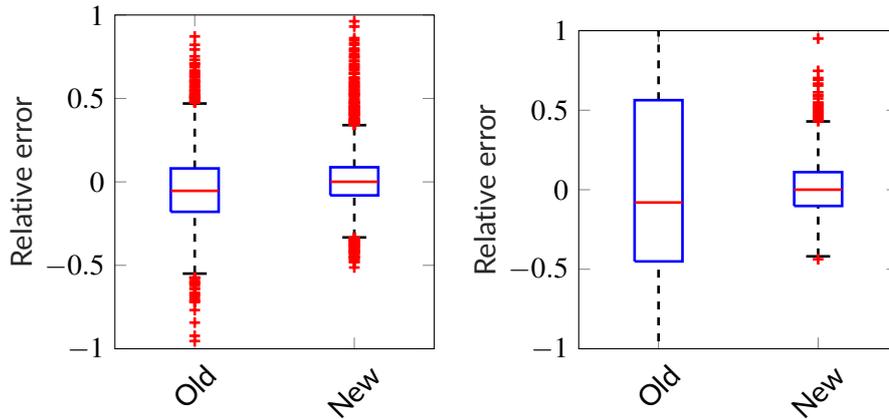
To check the validity of this approach, synthetic waves were generated using the centralized non-normalized lognormal function

$$f(x) = \exp\left(-\frac{(\ln(x))^2}{2\sigma_{ln}^2}\right). \quad (5.4)$$

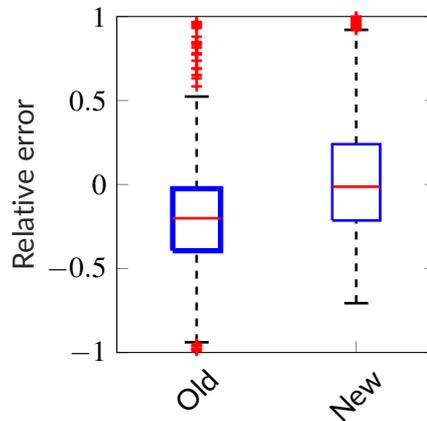
Varying σ_{ln} in the interval between 0.1 and 0.9 resulted in a set of curves with different upslopes, peakednesses and offsets. After flipping the curve in time and adding constant values in the beginning, the model functions in Figure 5.6(a) were obtained. The new feature algorithm calculating upslope, peakedness and ST changes was applied to every single wave. Figure 5.6(b)-(d) show the normalized feature values dependent on σ_{ln} . All feature values change monotonously as expected.

5.2.1.4 Discussion and Conclusion

In this study, the effect of noise on the feature extraction was quantified. Following that, existing algorithms were improved regarding their robustness. As already explained, a ground truth value for a feature value is often hard to determine and there is no standard



(a) Approaches for the calculation of T downslope. (b) Approaches for the calculation of T peakedness.



(c) Approaches for the calculation of T upslope.

Figure 5.5: Comparison of improved feature calculation methods with the former implementations as shown in Figure 5.2.

database including annotated feature values. However, a proof for robustness with simulated ECGs is possible. Although it was checked if the feature algorithms deliver the same values in the case of noise-free data and noisy data, this does not mean that the feature value itself is correct. There is no unique definition for a feature value e.g. describing the slope of a wave. To tackle this problem, the reversed lognormal function was utilized. The feature extraction algorithm was expected to deliver decreasing feature values for peakedness and upslope and increasing values for the ST change with increasing σ_m . Nevertheless, this plausibility check again does not include a ground truth value for the respective features, since there is a lack of an exact feature definition. A further point for discussion is the fixed SNR. Although, different sources confirm that this is a realistic choice [76–78], the value can be even lower. However, the improvement of the SNR should be part of a preceding processing step.

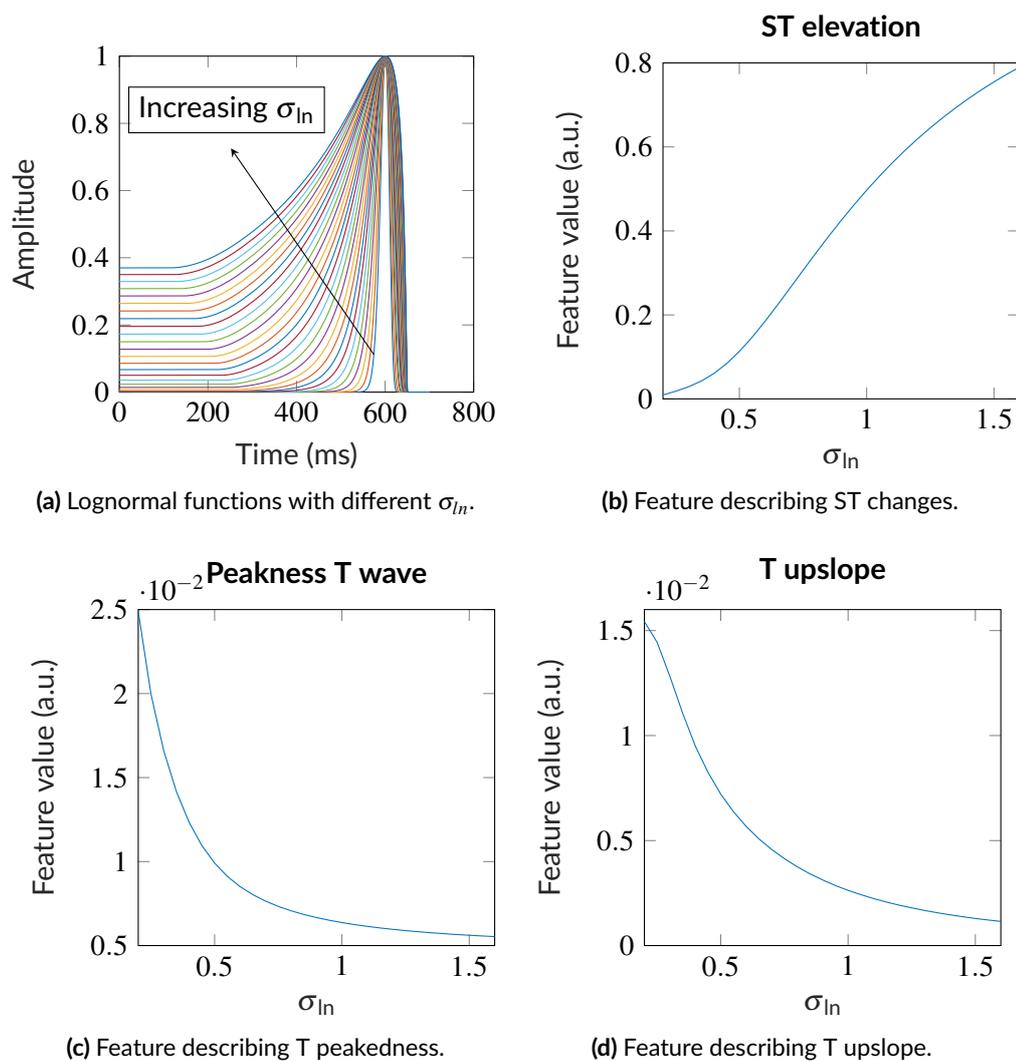


Figure 5.6: Experiment with different lognormal curves shown in (a). Three features were extracted (b)-(d) showing the changes as expected.

In conclusion, simulated signals without and with added noise serve as an excellent tool to evaluate the robustness of feature extraction techniques. Methods can be evaluated, compared and optimized for the application to (noisy) clinical data. Furthermore, inaccuracies can now be quantified which helps to explain possible discrepancies during clinical application.

5.2.2 Influence of Lead Reduction Techniques

5.2.2.1 Motivation

Having a look at Table 4.1, it gets apparent that different established methods used different lead combinations, i.e. selected leads from the 12-lead ECG, transformed leads from a principal component analysis (PCA) or the whole 12-lead ECG in the case of convolutional neural networks (CNNs). The justification of this choice is lacking in all these works and the question remains how this choice influences the final regression result. In the following study, this influence was quantified using a simple regression method and pre-selected features. *The study is based on the results from the student research project by María Hernández Mesa which were published with a similar methodology in [79] This study here is based on a larger database and the improved feature extraction techniques.*

5.2.2.2 Methods

A dataset of 81 noisy and noise-free simulated 12-lead ECGs was used as described in Section 5.2.1. The PCA coefficients for the first and second component of the whole beat were calculated, as well as the PCA coefficients (for only the first component) of the T wave part. Both sets of coefficients were applied to the single beat signals yielding two different transformed signals, in the following named $\text{PCA}_{\text{RT}} 1^{\text{st}}$, $\text{PCA}_{\text{RT}} 2^{\text{nd}}$ and PCA_{T} . Furthermore, a transform not maximizing the variance of the T wave as with PCA_{T} but the amplitude of the T wave was calculated (MaxAmp). The resulting transformation coefficients were applied to the single beat signals again. Hence, 15 leads in total were available in the study. These were used to estimate $[\text{K}^+]_{\text{b}}$ and $[\text{Ca}^{2+}]_{\text{b}}$ from features using linear regression with a polynomial of first and third order lead by lead. The fit for the third order polynomial was regularized with Tikhonov first order. Noise-free as well as noisy ECGs (filtered according to the findings from Section 5.2.3) were evaluated. Features used for $[\text{K}^+]_{\text{b}}$ estimation were: T peakedness, T wave amplitude, T wave up-/downslope, T ratio 1st half and R peak amp. Features considered for $[\text{Ca}^{2+}]_{\text{b}}$ estimation were T center, ST elevation and RT distance. The feature choice was based on findings from [62, 75, 80]. The improved feature extraction algorithms were used as described in Section 5.2.1.

Table 5.2: Estimation errors for $[K^+]_b$ and $[Ca^{2+}]_b$ on noise-free and noisy data using a first order and third order polynomial fit. Errors are given as mean \pm standard deviation of signed errors in mmol/l. The single lead showing the lowest error among all of the 12-lead ECG is given in brackets in the order of the columns.

1st order fit	Error $[K^+]_b$	Error $[Ca^{2+}]_b$	Error $[K^+]_b$ noisy	Error $[Ca^{2+}]_b$ noisy
MaxAmp	0.00\pm0.10	-0.00 \pm 0.18	0.00 \pm 0.37	-0.00\pm0.20
PCA _{RT} 1 st	0.00 \pm 0.43	0.01 \pm 0.32	-0.00 \pm 0.79	-0.00 \pm 0.35
PCA _{RT} 2 nd	0.00 \pm 0.11	0.00 \pm 0.22	0.00\pm0.35	0.00 \pm 0.25
PCA _T	-0.01 \pm 0.23	-0.00 \pm 0.18	0.00 \pm 0.42	-0.00 \pm 0.21
Best lead (II/V6/V6/V6)	-0.00 \pm 0.17	-0.00\pm0.12	-0.00 \pm 0.61	-0.00 \pm 0.21
3rd order fit	Error $[K^+]_b$	Error $[Ca^{2+}]_b$	Error $[K^+]_b$ noisy	Error $[Ca^{2+}]_b$ noisy
MaxAmp	0.00 \pm 0.06	0.00\pm0.09	0.01 \pm 0.38	0.01\pm0.16
PCA _{RT} 1 st	-0.13 \pm 1.43	-0.05 \pm 0.46	-0.00 \pm 2.16	0.00 \pm 0.34
PCA _{RT} 2 nd	0.00\pm0.04	-0.00 \pm 0.10	0.01\pm0.36	-0.00 \pm 0.22
PCA _T	0.01 \pm 0.15	0.00\pm0.09	0.01 \pm 0.41	0.01 \pm 0.17
Best lead (aVR/V1/III/V6)	-0.00 \pm 0.10	-0.00\pm0.09	-0.00 \pm 2.16	0.00 \pm 0.17

5.2.2.3 Results

Lead reduction influenced in both the noise-free and noisy case the estimation result. As visible in Table 5.2, lead reduction led to a worse result than the processing of the best single lead in the noise-free case for $[Ca^{2+}]_b$ estimation for a first order fit and an equal result for a third order fit. Nevertheless, results improved for $[K^+]_b$ estimation. This changed when noise was added to the signal: the use of lead transform (except PCA_{RT} 1st) delivered better results than using the single leads. Apart from this fact, it got apparent that between the lead reduction techniques, differences were visible, too. PCA_{RT} 2nd and MaxAmp delivered the most reliable results for all experiments. For $[Ca^{2+}]_b$ estimation alone, PCA_T led to results comparable to the two aforementioned methods.

5.2.2.4 Discussion and Conclusion

This study was aiming at quantifying the influence of the choice of a lead reduction technique. This was evaluated with two regression techniques using preselected features. In the case of $[K^+]_b$ estimation, PCA_{RT} 2nd, PCA_T and MaxAmp delivered the best results on noisy ECGs. The noise reducing characteristics of these transformations are beneficial for the estimation. PCA_{RT} 1st showed in all settings the worst results as the transformation is mainly influenced by the R peak. This can lead to T waves with small amplitudes in the resulting signal making feature extraction more complicated (Figure 5.8). This is substantiated by the fact that transformations specifically aiming at maximizing the influence of the T wave, delivered better results. In the noise-free setting, PCA_{RT} 2nd and MaxAmp delivered the best results, for a third order fit even better results than for the first order fit. On the one hand, this might be connected to a better fitting model for the concentration feature behavior, and on the other hand, to the influence of the transformation on exactly this concentration feature behavior. Figure 5.7 shows the connection between feature and concentration value

for selected lead transformations and the best single lead which is of course not known in advance. While leads II and V6 showed a nearly linear behavior, the behavior for lead transformation follows more a quadratic (or higher order) relationship.

For $[Ca^{2+}]_b$ estimation, $PCA_{RT} 2^{nd}$, PCA_T and MaxAmp also delivered the best results. In the noise-free case, MaxAmp and PCA_T delivered errors as low as the best single lead. It seems that these transformations preserve the information being important for the concentration estimation based on the selected features. In conclusion, the selection of appropriate lead reduction techniques is essential for the estimation of ionic concentrations. In general, $PCA_{RT} 2^{nd}$, PCA_T and MaxAmp are promising candidates for a concentration estimation.

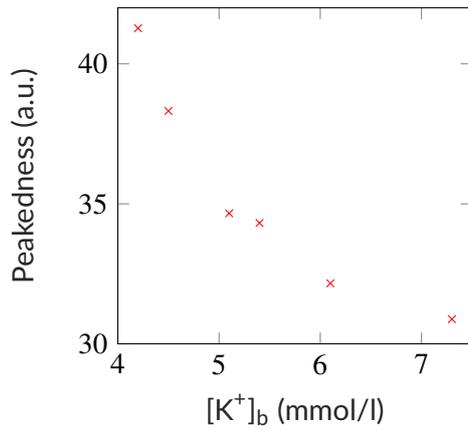
5.2.3 Influence of Filtering on Feature Extraction

5.2.3.1 Motivation

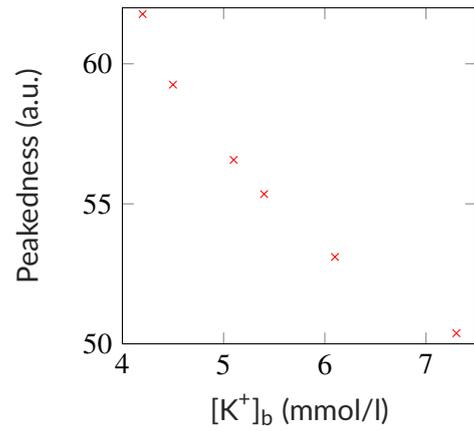
Even the most robust feature extraction technique depends on the input. If the information is overlaid with noise, signal processing methods can help to separate the noise from the signal of interest. Nevertheless, if a pre-processing step (like the usual bandpass filtering) already distorts the information, even the most advanced method cannot reconstruct it. A trade-off exists between attenuating the noise and not disturbing the signal parts of interest. In the following, the optimal bandpass filtering limits will be evaluated in terms of not influencing the feature extraction while still attenuating noise as much as possible.

5.2.3.2 Methods

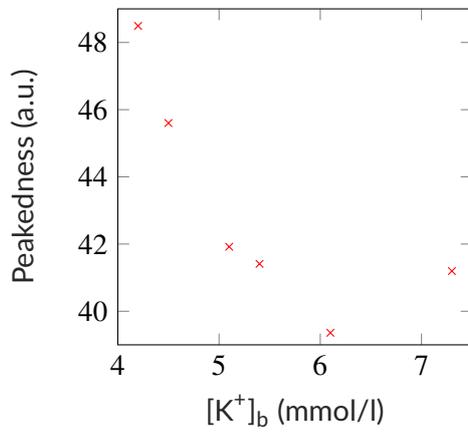
The 81 simulated 12-lead ECGs were superimposed by white Gaussian noise (0 dB, 3 dB, 10 dB, 20 dB, 30 dB) and bandpass filtered with different cut-off frequencies for high-pass and low-pass. This was repeated for each signal 20 times using different realizations of the noise. Additionally, the noise-free signals were filtered with the same cut-off frequencies. The features as presented before were extracted and the relative error compared to the noise-free unfiltered features were calculated for both the noisy filtered and noise-free filtered signals. Comparing the results between noise-free filtered and noise-free unfiltered helps to find a lower bound for the low-pass filter. Here, the influence of the filtering on the signal itself was evaluated. The comparison between noisy filtered and noise-free unfiltered helps to find the upper bound for the cut-off frequency of the low-pass. It was assumed that there is one cut-off frequency at which the result does not change any more with increasing cut-off frequencies meaning that only noise was filtered. This holds similarly for the high-pass filter but estimating an upper instead of a lower bound with the noise-free filtered signals.



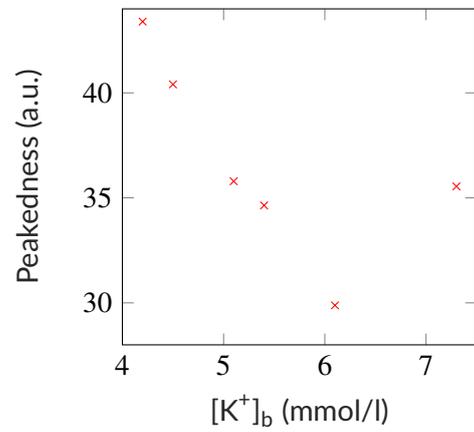
(a) Feature concentration behavior for T peakedness calculated on **II**.



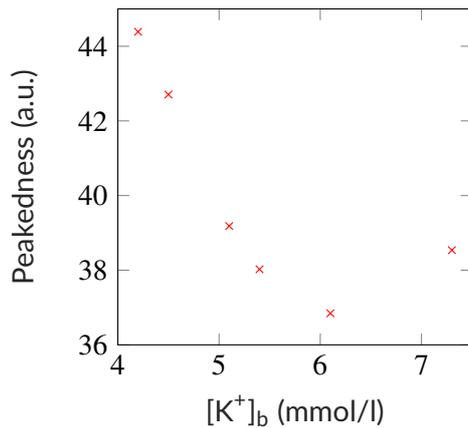
(b) Feature concentration behavior for T peakedness calculated on **V6**.



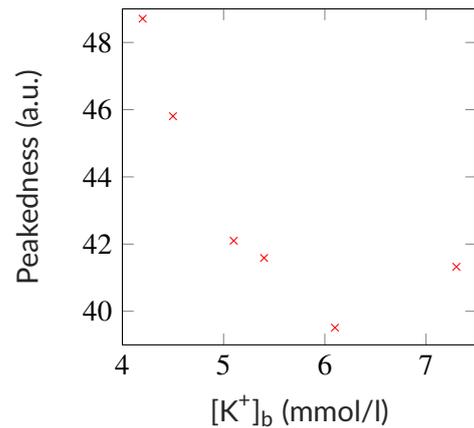
(c) Feature concentration behavior for T peakedness calculated on **MaxAmp**.



(d) Feature concentration behavior for T peakedness calculated on **PCA_{RT} 1st**.



(e) Feature concentration behavior for T peakedness calculated on **PCA_{RT} 2nd**.



(f) Feature concentration behavior for T peakedness calculated on **PCA_T**.

Figure 5.7: Feature $[K^+]_b$ behavior for four lead transformed ECGs and two standard leads. $[Ca^{2+}]_b$ was constant.

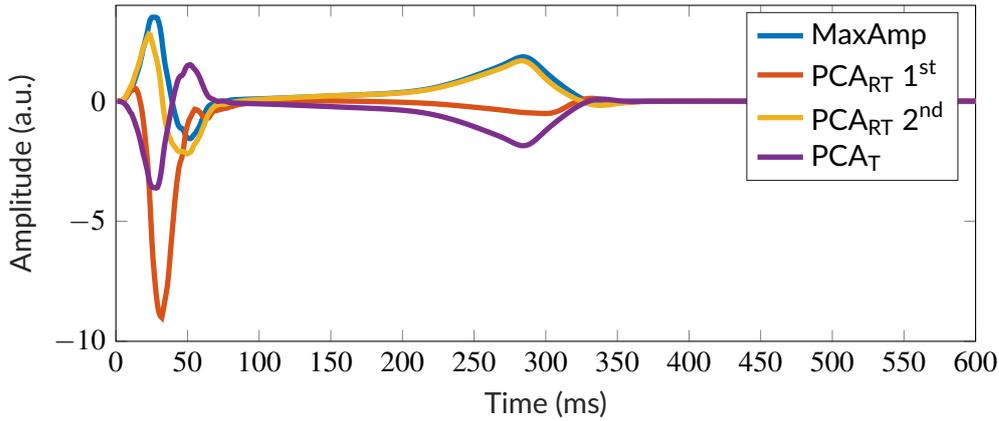


Figure 5.8: Transformed ECG signals. $\text{PCA}_{\text{RT}} 1^{\text{st}}$ shows lowest T wave amplitude.

Table 5.3: Averaged (over all features) absolute values of median relative errors for different cut-off frequencies for the low-pass filter. The high-pass cut-off frequency was 0.05 Hz. f_c : cut-off frequency.

f_c	20 Hz	40 Hz	50 Hz	60 Hz	70 Hz	80 Hz	100 Hz	150 Hz	200 Hz
Noise-free	0.055	0.005	0.002	0.002	0.001	0.001	0.001	0.001	0.001
Noisy	0.054	0.015	0.014	0.013	0.014	0.015	0.018	0.028	0.037

5.2.3.3 Results

In the following, only the results for an SNR of 20 dB will be shown for the sake of clarity. Similar results were obtained with the other SNR values and are visible in Section A.2. Results are visualized in Figure 5.9 to Figure 5.12. Here, the median errors (line plots) and the interquartile range (bars) of the relative errors of the feature extraction (related to the unfiltered feature values) for all of the T wave features and selected cut-off frequencies for high-pass and low-pass are shown. Figure 5.9 and Figure 5.11 show the errors for the noise-free filtered data and Figure 5.10 and Figure 5.12 for the noisy input data. The smallest median relative errors averaged over all features were achieved for a low-pass cut-off frequency between 60 Hz and 100 Hz for the noisy data (Table 5.3). Figure 5.10 underlines this finding. In this interval, the median relative errors were below 5% (red dashed lines). The averaged median relative errors remained quasi constant from 60 Hz on for the noise-free filtered errors (Figure 5.10). For the high-pass filtering, the highest acceptable averaged median value for noise-free and noisy data was found to be at 0.1 Hz (Table 5.4). The feature being mostly influenced by high-pass filtering was ST change (Figure 5.11).

5.2.3.4 Discussion and Conclusion

The aim of this investigation was to assess the influence of bandpass filtering on the feature extraction. From the results, borders for an adequate filtering can be extracted. A note of

Table 5.4: Averaged (over all features) absolute values of median relative errors for different cut-off frequencies for the high-pass filter. The low-pass cut-off frequency was 60 Hz. f_c : cut-off frequency.

f_c	0.05 Hz	0.1 Hz	0.2 Hz	0.3 Hz	0.4 Hz	0.5 Hz
Noise-free	0.002	0.001	0.001	0.001	0.003	0.005
Noisy	0.013	0.012	0.018	0.024	0.031	0.031

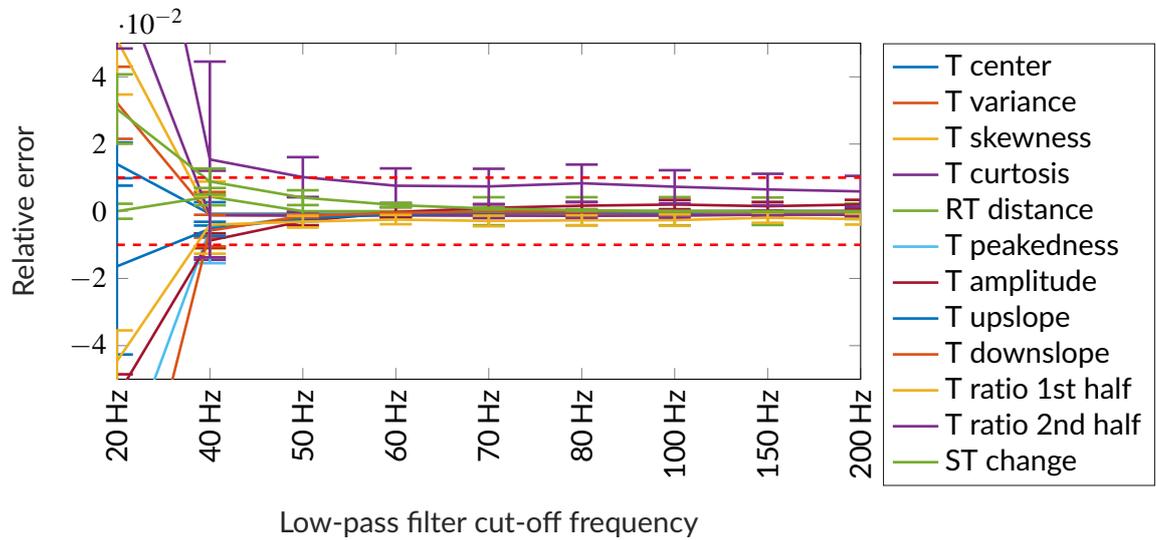


Figure 5.9: Relative errors of T wave feature extraction algorithms for selected low-pass cut-off frequencies. The algorithms were applied on the noise-free filtered data and compared with the features extracted from the unfiltered noise-free data. The lines represent the median values of the relative errors, the bars visualize the interquartile ranges. HP: high-pass, LP: low-pass; all values on the x-axis are cut-off frequencies in Hz. The red dashed lines mark errors of $\pm 1\%$.

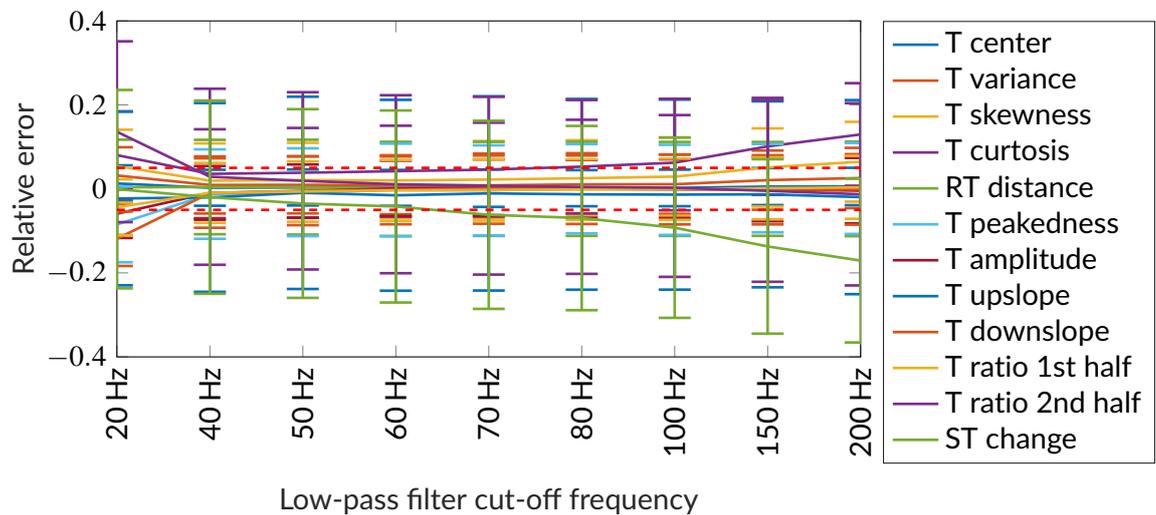


Figure 5.10: Relative errors of T wave feature extraction algorithms for selected low-pass cut-off frequencies. The algorithms were applied on the noisy filtered data and compared with the features extracted from the unfiltered noise-free data. The lines represent the median values of the relative errors, the bars visualize the interquartile ranges. HP: high-pass, LP: low-pass; all values on the x-axis are cut-off frequencies in Hz. The red dashed lines mark errors of $\pm 5\%$.

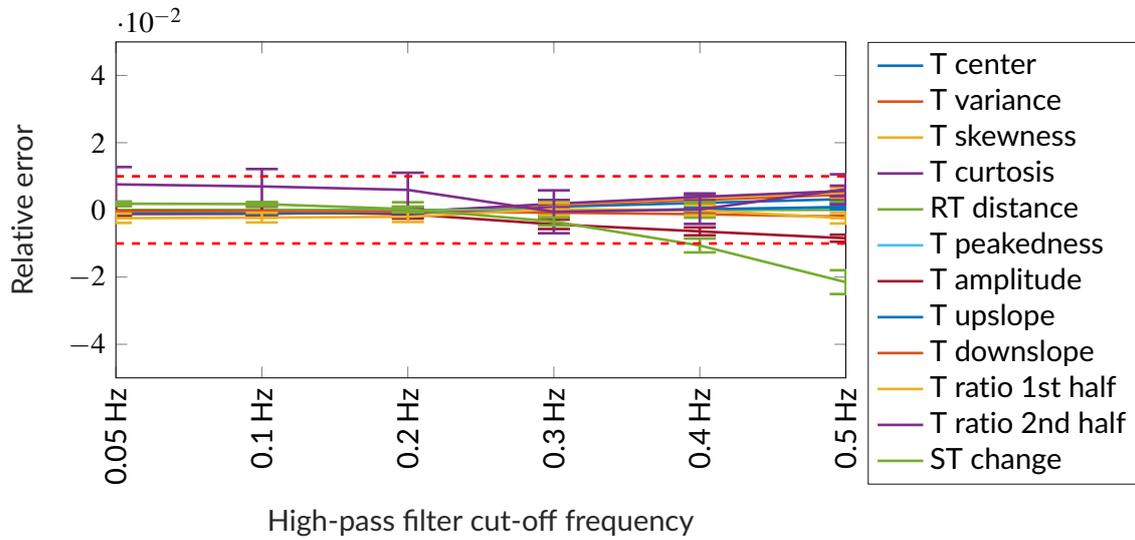


Figure 5.11: Relative errors of T wave feature extraction algorithms for selected high-pass cut-off frequencies. The algorithms were applied on the noise-free filtered data and compared with the features extracted from the unfiltered noise-free data. The lines represent the median values of the relative errors, the bars visualize the interquartile ranges. HP: high-pass, LP: low-pass; all values on the x-axis are cut-off frequencies in Hz. The red dashed lines mark errors of $\pm 5\%$.

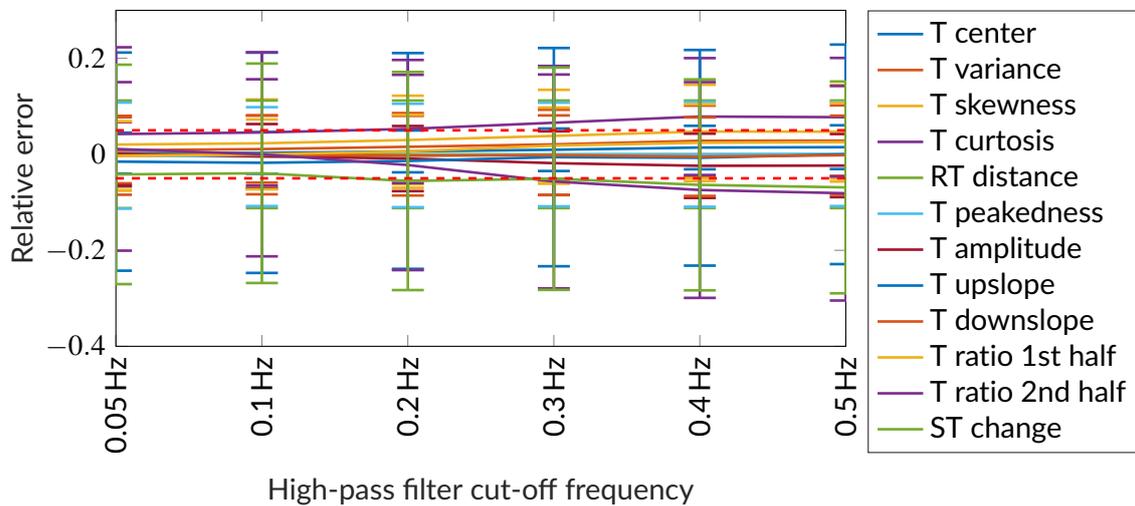


Figure 5.12: Relative errors of T wave feature extraction algorithms for selected high-pass cut-off frequencies. The algorithms were applied on the noisy filtered data and compared with the features extracted from the unfiltered noise-free data. The lines represent the median values of the relative errors, the bars visualize the interquartile ranges. HP: high-pass, LP: low-pass; all values on the x-axis are cut-off frequencies in Hz. The red dashed lines mark errors of $\pm 5\%$.

caution is due here since the presented results are only valid assuming an SNR of 20 dB and for exactly these feature extraction algorithms. However, one should keep in mind, that ECGs with an SNR of 20 dB are common to measure, especially in resting patients [76–78]. Another crucial point was again highlighted in this study: Although it is suggested in literature, that the spectral contents of the T wave are located up to around 10 Hz, it was clearly visible, that a low-pass filter with a cut-off frequency of 20 Hz (and still 40 Hz) clearly distorts most of the feature values (Figure 5.9). In conclusion, the bandpass filtering step should include a high-pass filter with a cut-off frequency of 0.1 Hz and a low-pass filter with at least 60 Hz or more assuming an SNR of 20 dB.

5.3 Proof of Concept Study for Calcium and Potassium Concentration Estimation

5.3.1 Motivation

After the optimization of the pre-processing steps, a final evaluation of the estimation of the concentrations is still missing. Although reconstruction results were already presented in the last sections, features were selected empirically and the used methods remained simple. In this study, a systematic feature selection method was used and more advanced methods for reconstruction were utilized. Furthermore, the influence of noise on the reconstruction result was evaluated systematically. *This study was similarly published in [81].*

5.3.2 Methods

The 81 simulated ECGs from the Himeno model were used during this study. Features were extracted from a transformed lead using the MaxAmp transformation. The transformation matrix was calculated for the reference concentration ($[\text{Ca}^{2+}]_o=1.8$ mmol/l and extracellular potassium concentration ($[\text{K}^+]_o=4.5$ mmol/l) and applied to all ECGs. To reduce the number of features from 16 to six, Canonical Correlation Analysis (CCA) was used [82]. CCA quantifies redundancy by correlating one feature with a linear combination of all others. The analyzed feature was dropped if it was redundant in terms of showing a high correlation with the others. This was repeated until the desired number of six features was achieved.

For estimating $[\text{K}^+]_b$ and $[\text{Ca}^{2+}]_b$, a shallow neural network was selected as regression method. The small amount of data made regularization essential: a regularized Bayesian neural network with early stopping was used as implemented in MATLAB's Machine Learning Toolbox (MATLAB 2020b, The MathWorks, Inc., Natick, Massachusetts, United States). The early stopping criterion was evaluated with six randomly chosen training samples. The network consisted of one hidden layer with 12 neurons. Random initialization of the weights was repeated 20 times to exclude possible effects of a bad initialization.

The estimation study was performed with two different underlying datasets: First, using the 81 noise-free simulated ECGs, second, with 81 noisy ECGs (SNR of 20 dB) and third with 81x20 noisy ECGs, i.e. 20 different realizations of noise (SNR of 20 dB) were added to each of the 81 simulated ECGs. The noisy data were prefiltered with a low-pass of 70 Hz and a high-pass of 0.1 Hz according to Section 5.2.3. Leave-one-out cross validation was selected as validation method. In the augmented dataset, it was ensured that noisy ECGs coming from the same noise-free ECG were always either in the validation or training dataset. The concentration distribution in the simulated dataset was 5.15 ± 1.00 mmol/l for $[K^+]_b$ and 1.78 ± 0.74 mmol/l for $[Ca^{2+}]_b$.

5.3.3 Results

CCA determined R amplitude, T center, RT distance, T peakedness, T amplitude, ST change as the six most non-redundant features. The results of the estimation with these features are summarized in Table 5.5. Mean and standard deviations were calculated for all 20 random initializations of the neural network and then averaged. As expected, concentration estimation on noise-free data performed better than on noisy data. Using the six features, a standard deviation (signed errors) of 0.06 mmol/l was achieved for $[Ca^{2+}]_b$ estimation, 0.07 mmol/l for $[K^+]_b$ estimation. In the noisy, non-augmented case, the standard deviation of the errors increased to 0.74 mmol/l, respectively 0.79 mmol/l. With the augmentation, the estimation performance again improved to a standard deviation of 0.1 mmol/l for $[Ca^{2+}]_b$ and 0.16 mmol/l for $[K^+]_b$. In addition to the signed mean and standard deviations in Table 5.5, unsigned errors were also evaluated to make the study comparable to other published articles.

Table 5.5: Calcium/Potassium estimation error in mmol/l given as mean \pm standard deviation for signed and unsigned errors.

$[Ca^{2+}]_b$	signed	unsigned	$[K^+]_b$	signed	unsigned
Noise-free	0.00 ± 0.06	0.03 ± 0.05	Noise-free	0.00 ± 0.07	0.04 ± 0.06
Noisy 1 rep.	-0.01 ± 0.74	0.64 ± 0.38	Noisy 1 rep.	-0.03 ± 0.79	0.62 ± 0.50
Noisy 20 rep.	-0.00 ± 0.10	0.07 ± 0.07	Noisy 20 rep.	0.00 ± 0.16	0.12 ± 0.10

5.3.4 Discussion and Conclusion

In contrast to the before used approaches for concentration estimation, feature selection was this time not performed empirically but in a more objective way. Although CCA considers only linear dependency as selection criterion and this does not per se imply the suitability of the feature, all features were designed for the concentration estimation. Nevertheless, the

number of features was empirically chosen and the influence on the result was not evaluated here.

Comparing the standard deviations of errors with the standard deviations of concentrations in the dataset, especially the method trained on noise-free and on the augmented noisy case performed well. However, without augmentation, the standard deviation of the estimation error were quite high. The neural network is not capable of learning to distinguish between noise and actual feature value changes if only one noise realization is considered. Nevertheless, when passing more information by artificially augmenting the dataset, the network learns to correct the uncertainty in the feature values. This was expected since the regression method is then able to average over different realizations of a zero mean noise. Another option to decrease the estimation error could be to use more and possibly redundant features.

Different regularization techniques for neural networks (e.g. a Bayesian neural network) were utilized to prevent overfitting together with a low network complexity with only one hidden layer with 12 neurons in it. As there is no standard measure to quantify overfitting, the errors in the training, validation, test dataset were compared. Figure 5.13 visualizes the errors in the single datasets. It is visible that - as expected - the training dataset is the one with the lowest error. Nevertheless, errors in the test dataset are in a comparable range. Thus, overfitting should not be present.

In conclusion, the estimation of both $[Ca^{2+}]_b$ and $[K^+]_b$ seems to be feasible with the proposed features and methods. For a noise-free signal, mean absolute errors as low as 0.03 mmol/l can be achieved. Nevertheless, an optimized pre-processing or redundant data is needed to compensate the influences of noisy data. It should be emphasized that these results were generated on one geometry. The influence of different heart geometries on the features still needs to be evaluated.

5.4 Influence of Patient Geometry

5.4.1 Motivation

Until now, the influences and feasibility of concentration estimation with only one ventricular geometry located in one specific torso were analyzed. However, when applying these methods to clinical data, the question arises if a general model is still valid or if patient-specific properties lead to different feature concentration behaviors that later need to be compensated. To quantify the influence, a study was performed varying the following properties:

- $[K^+]_o$ and $[Ca^{2+}]_o$
- Position of the heart in the torso
- Geometry of the ventricles

In the end, T wave feature values were compared between the selected anatomical and physiological variations helping to answer the above stated question. A comparable study was

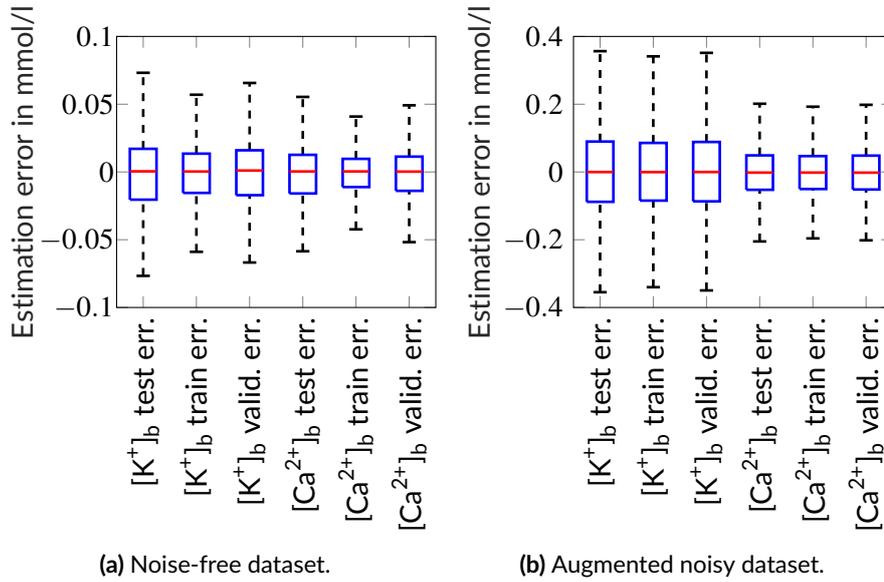


Figure 5.13: Errors from the training (train), validation (valid., for checking the early stopping criterion) and test dataset for $[K^+]_b$ and $[Ca^{2+}]_b$ estimation. Err.: error. Outliers are not visualized.

published in [83], but evaluating only the QRS complex. *This project was similarly presented in the Master thesis of Michael Meinzer [84] but now uses optimized algorithms, and it was extended by further geometrical variations. Michael Meinzer's work was supported by Claudia Nagel.*

5.4.2 Methods

To achieve a large number of simulations in a short time, monodomain simulations were not feasible. Instead, a combination of single cell simulations and activation patterns from the Eikonal approach were used to obtain a realistic ECG as similarly proposed in [85]. By doing so, it was possible to introduce geometrical, rotational and concentration variations. In the following, the underlying methodology will be explained, starting with the simulation method. Afterwards, the used optimization procedure in order to get a realistic T wave will be presented. Last, the parameter variations will be introduced.

5.4.3 Action Potential Simulation

APs were simulated using the extended Himeno et al. model parameterized with different ionic concentrations. Therefore, single cell simulations with a length of 60 s were performed triggering an AP every second using an existing simulation tool `ElphyModelTest`. Only the last AP was selected for further processing to exclude non-steady state APs.

5.4.4 Eikonal Activation Patterns and ECG Extraction

Activation patterns on a ventricular geometry were simulated using the Eikonal approach. Therefore, the location of seven trigger points for the activation were taken from [86]. As they are only given on a specific geometry, they were visually transferred to the other geometries. Ventricular fibers were calculated with an extended and adjusted approach available at the Institute of Biomedical Engineering based on [87]. Instead of a Purkinje tree, a fast conducting sub-endocardial layer ($cv_{fast} = 2000$ mm/s in fiber direction) on the endocardium was used to initiate the depolarization. The remaining nodes were assigned the conduction velocity $cv_{tis} = 650$ mm/s in fiber direction. The anisotropy factor between the conduction velocity in fiber direction and perpendicular to it was 1.64. The resulting activation times were subsequently used to shift the simulated APs in time. The obtained transmembrane voltage (TMV) distributions on the heart were blurred with a Laplacian filter kernel and forward calculated using spatially downsampled surfaces of the ventricles and of the torso surface [88]. The 12-lead ECG was extracted using the potentials on the torso surface.

5.4.5 Geometrical Variations

Three different ventricular geometries were used. They were generated with an adjusted version [89] of the ventricular statistical shape model presented in [90]. The torso geometry was not varied.



Figure 5.14: The three ventricular geometries used in the study. They were all generated with the shape model described in [90].

5.4.6 Optimization of the ECG Signal

With the methodology described until now, it is not possible to create a physiological ECG. Rather, a tuning of different parameters needs to be performed. In this study, the following parameters were optimized:

- the position change in apicobasal (ab) and rotational (rt) direction on the ventricular surface (Figure 5.15) of the seven trigger regions (their positions were only roughly known on the specific geometry),
- a delay time of the single trigger points,
- the conduction velocities cv_{fast} and cv_{tis} .

The parameter search was constrained as shown in Table 5.6. The optimization was performed with a genetic algorithm (100 generations, 24 individuals).

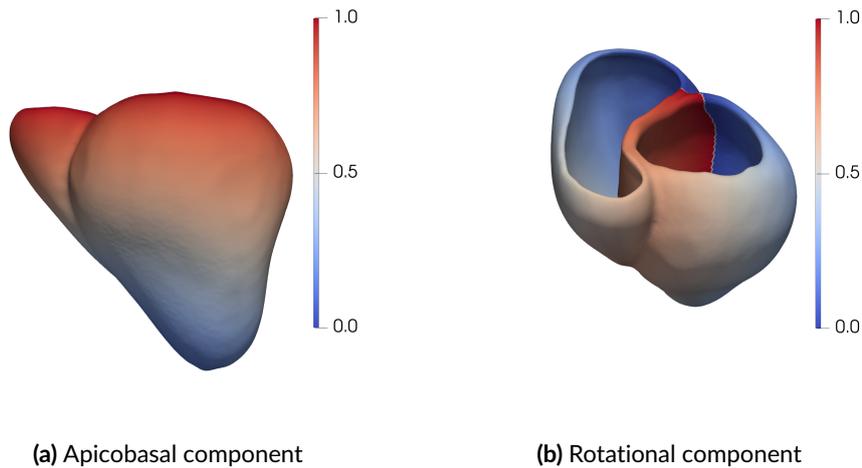


Figure 5.15: Apicobasal (ab) and rotational (rt) component on the geometry 1. The rotation is defined for each ventricle separately. Both are normalized between 0 and 1. The components were calculated with an adjusted version of [91].

To obtain a physiological T wave, further parameters were optimized (as similarly done in [69]):

- the conductivity of the I_{Ks} channel on the base $g_{Ks,b}$,
- the conductivity of the I_{Ks} channel at the apex $g_{Ks,a}$,
- a simultaneous scaling of both by the same factor.

The g_{Ks} values between apex ($g_{Ks,a}$) and base ($g_{Ks,b}$) were exponentially varied in apicobasal direction. Therefore, 20 discrete layers with constant $g_{Ks,b}$ values were defined. The

Table 5.6: QRS complex optimization constraints and initial values for the genetic algorithm. Position changes are given in percentages of the apicobasal (ab) length and the circumference (rt).

Parameter	min	init	max
cv_{fast} (mm/s)	1300	2000	2700
cv_{tis} (mm/s)	400	650	1200
delay (ms)	0	0.003-0.005	0.016
position change ab	-10%	0	10%
position change rt	-10%	0	10%

exponential change of g_{K_s} corresponds to a linear change of AP duration as described in [69]. The optimization of the T wave was again performed with a genetic algorithm (98 generations, 12 individuals). The constraints are given in Table 5.7.

Table 5.7: T wave optimization constraints and initial values for the genetic algorithm. $g_{K_{s,a,org}}$ and $g_{K_{s,b,org}}$ describe the original values.

Parameter	min	init	max
$g_{K_{s,b}}/g_{K_{s,b,org}}$	0.5	1	1.5
$g_{K_{s,a}}/g_{K_{s,a,org}}$	1.5	2.25	9.5
scaling	0.5	1	2.5

The optimization was always performed for the non-rotated geometry using the AP obtained with the default concentration proposed in the Himeno et al. model [33].

The fitness function for the genetic algorithm was calculated in the following way: First, the maximum value from a normalized cross-correlation was calculated between the template $x_t(l)$ and the result from the simulation $x_s(l)$ for every lead $l = 1 \dots 12$. The values were inverted and subtracted from one yielding the fitness function

$$\hat{f}_i = 12 - \sqrt{\sum_{l=1}^{12} \frac{\max(\text{cc}(x_t[l], x_s[l]))}{\sqrt{\max(\text{cc}(x_s[l], x_s[l]) \cdot \max(\text{cc}(x_t[l], x_t[l]))}}}}. \quad (5.5)$$

cc is the cross-correlation function. The denominator guarantees that the subtrahend is maximum one. For the optimization of the QRS complex, x_t and x_s contained only the QRS complex, for T wave optimization, the T wave respectively. As ground truth templates, three ECG signals of healthy subjects from the PTB diagnostic database on physionet [92, 93] (s0292lre, s0306lre, s0312lre), each for one geometry, were used.

5.4.7 Rotational Variations

Applying the found set of simulation parameters, the position of the heart in the torso was varied. Therefore, each ventricular geometry was rotated around four axes: first, an axis from the base to the apex in the left ventricle (rotation described by the angle γ_{roll}), second, a anterior-posterior axis (rotation described by the angle γ_{yaw}) and third, an axis perpendicular to the two before mentioned (rotation described by the angle γ_{pitch}). The axes are visualized

in Figure 5.16. According to [94], the standard deviation of these angles are between 8.8° and 12.2° . Considering this, a variation of approximately twice the mean standard deviation ($\pm 20^\circ$) was used for all angles changing them separately in steps of 2° from simulation to simulation. Furthermore, 100 different angle combinations (respecting the before mentioned borders) were randomly chosen from a uniform distribution to simulate a simultaneous angle change. By doing so, for each geometry, 163 rotational variations were created.

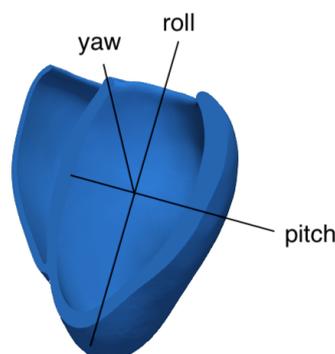


Figure 5.16: The three rotation axes for roll (γ_{roll}), yaw (γ_{yaw}) and pitch (γ_{pitch}) angle. Adapted from [84].

5.4.8 Concentration Variations

Next to the rotational variations, $[K^+]_o$ was varied in discrete steps: 3.8 mmol/l, 4.5 mmol/l, 6.3 mmol/l and 7.8 mmol/l. Furthermore, $[Ca^{2+}]_o$ was set to the following values: 0.6 mmol/l, 1.0 mmol/l, 1.8 mmol/l, 2.6 mmol/l and 3.0 mmol/l.

5.4.9 Results

For every geometry, in total 3260 12-lead ECGs were extracted with different underlying rotations and ionic concentrations. To reduce the amount of information to a manageable size, the MaxAmp transformation was used. Moreover, only results for T amplitude and T downslope will be presented in the following since these two features were the most important ones for the following study on clinical data (Chapter 6) and for most of the methods from literature (Table 4.1), the approach was compared to.

Figure 5.17 shows the results from feature calculation on the 100 rotational variations of geometry 1. It gets apparent that the boxes in both subplots representing absolute feature values are overlapping especially for lower potassium concentrations. This makes a patient-independent concentration estimation difficult as there is no unique relationship between concentration value and feature value. A similar behavior is visible in Figure 5.18, but with

overlapping values for higher concentrations. Similar findings were made on single leads and for other features.

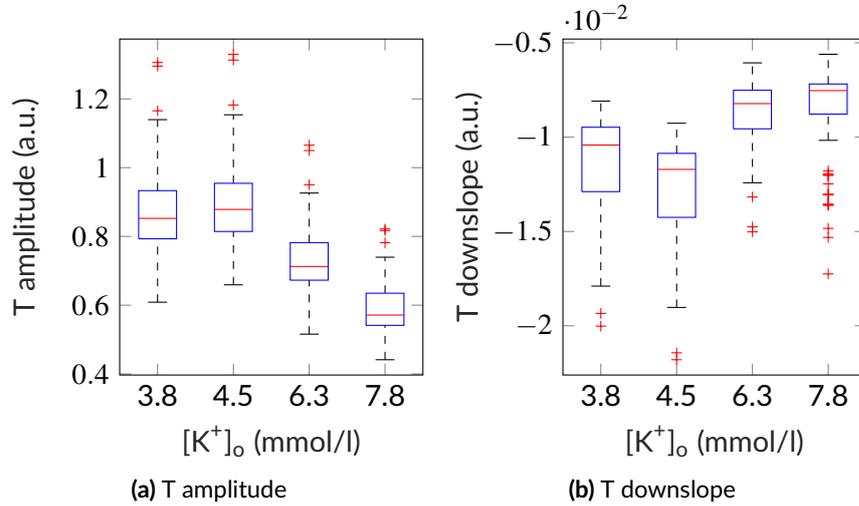


Figure 5.17: Feature values for different $[K^+]_o$ for the 100 random rotations of geometry 1.

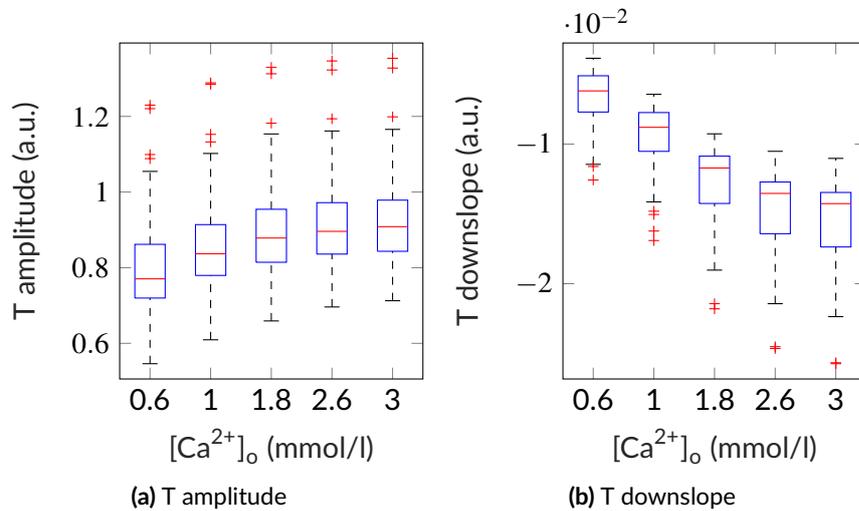


Figure 5.18: Feature values for different $[Ca^{2+}]_o$ for the 100 random rotations of geometry 1.

Comparing the results of different geometries with only varying one angle emphasizes the dependency of the results from the actual geometry. Figure 5.19 shows the results for T amplitude and T downslope for different γ_{roll} and the three geometries. Modifying γ_{roll} , feature values were changing, but the absolute change with a changing γ_{roll} was different

between the geometries. Similar results were visible for γ_{pitch} and γ_{yaw} (Figure A.9 and Figure A.10). This again underlines that the relation of feature values between different patients might change even with normal concentrations.

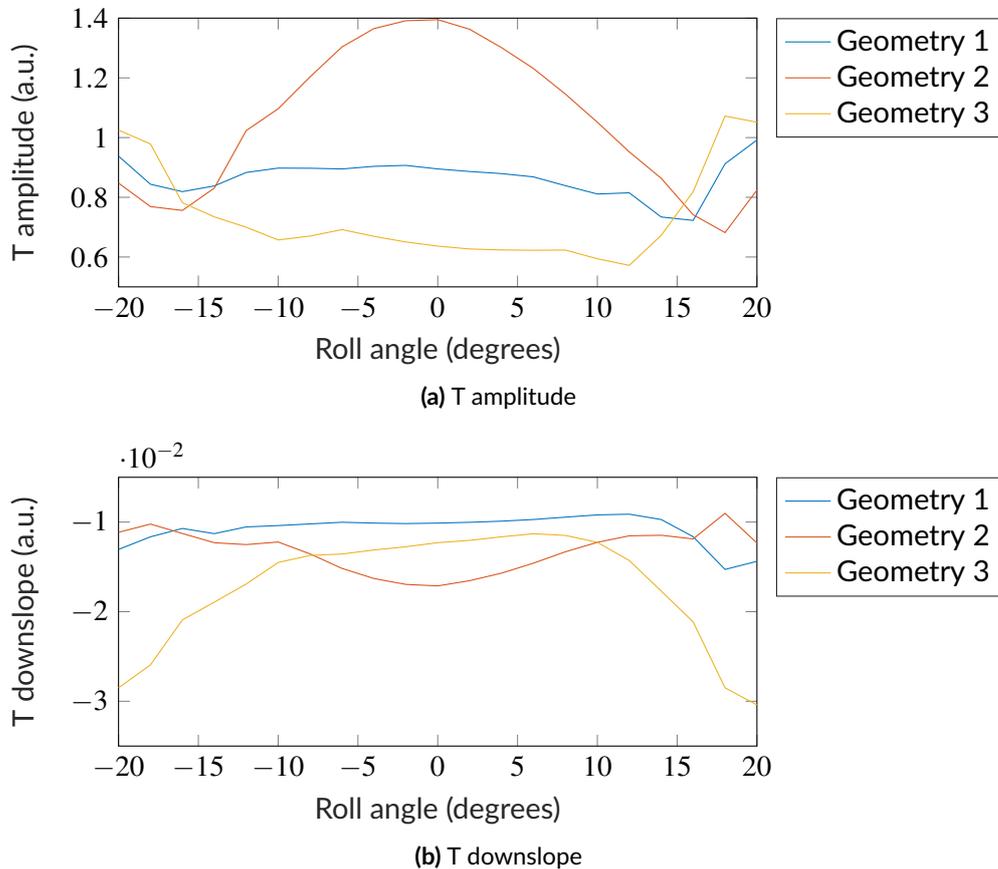


Figure 5.19: Feature changes dependent on γ_{roll} for the three geometries for the default concentration of the Himeno et al. model.

5.4.10 Discussion and Conclusion

To obtain an estimation of the dependency of the features from the patient geometry, a simplified simulation method was used to extract a large number of simulations in a short time. This is of course connected with certain shortcomings in the extracted ECGs. From this, the first topic to discuss arises: the realisticness of the simulations which was surely limited. Figure 5.20 shows two exemplary signals from a realization with the corresponding ECG template used for the optimization. Although it could be that the correspondence between template and simulations was low, for the final evaluation of the influence of the rotations, this should be of secondary importance since only changes were evaluated.

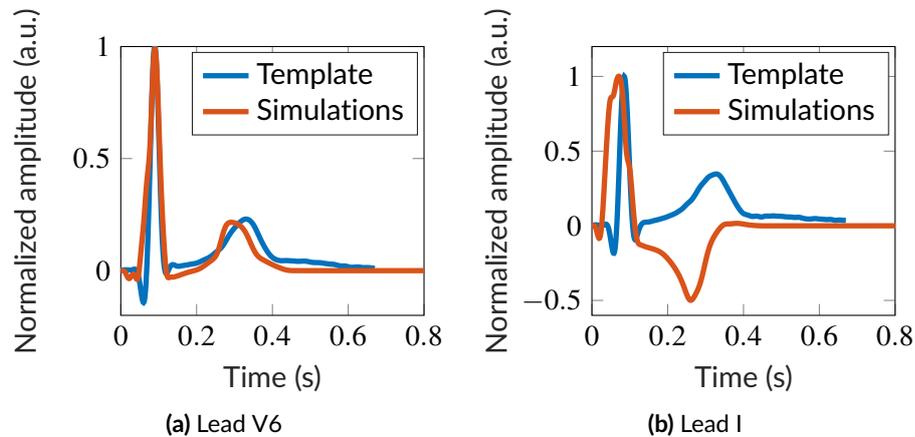


Figure 5.20: Comparison of simulated ECG signals with the templates. (a) shows an example for a good correspondence between simulation and template, (b) shows an example with a low correspondence.

Compared to a monodomain simulation, diffusion processes were completely neglected on the cellular level which also influence the shape of the ECG [95] and might be crucial for realistic ECGs. Figure 5.17(a) shows an unexpected progression of the feature for increasing $[K^+]_o$. Amplitudes are expected to rise with increasing $[K^+]_o$ like it was the case with the results from the monodomain simulation (Figure 5.21). This should be definitely considered for the interpretation of the results. It cannot be guaranteed that the shape (and therefore the absolute feature values) of the ECG waves would show a similar variation for different $[K^+]_o$ as with an underlying monodomain simulations. Nevertheless, the variations for a fixed concentration are quite large so an ambiguity between feature value and underlying concentration value is likely. Besides that, monodomain simulations show a relative T amplitude change between $[K^+]_o=3.8$ mmol/l and $[K^+]_o=6.7$ mmol/l of 84%, whereas relative feature variations caused by geometrical changes as shown in in Figure 5.19 were up to 42%. To sustain the fast computation and get more accurate results, the implementation and application of a Reactional-Eikonal model could be considered in a follow-up study as proposed in [95].

A further simplification was connected with the applied forward model. In this study, a homogeneous torso was used with the boundary element method for solving the underlying Poisson's equation. It is known, that the introduction of other organs into the torso causes ECG shape changes [96]. However, the use of an inhomogeneous torso does not guarantee a realistic ECG since there is no clarity how to parameterize the conductivities of the organs although they influence the resulting ECG severely [42]. In spite of this, for the evaluation of the influence of the rotation, mainly the change of a feature was relevant, so this simplification should be justified.

Regarding the optimization process with the genetic algorithm, further aspects for discussion emerge: The used template during optimization was taken from PTB diagnostic database on Physionet [92, 93] and naturally does not need to fit the geometrical shape model instance.

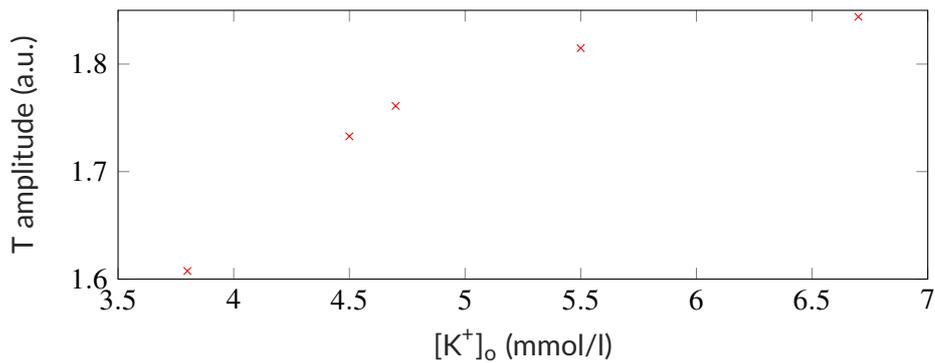


Figure 5.21: T amplitude feature values for different $[K^+]_o$ obtained from the monodomain simulations.

Moreover, the choice of seven trigger points according to [86] is not guaranteed to be valid for every individual. For example, Durrer et al. identified only four trigger points in the left and right ventricle [97]. Both the choice of the template and the number and locations of the trigger points leads to the problem that a perfect fit cannot be achieved. Furthermore, the used constraints were arbitrary as for the phenomenological modeling, it is hard to extract these meta parameters from measurements. Nevertheless, they were successfully applied in a preceding study [84]. Last, the number of iterations for the genetic algorithm was set to 100 due to computation time constraints. Better results might have been gained for more iterations.

In conclusion and apart from all discussed issues, this study underlines the patient-dependency of T wave features. These findings need to be considered for an application to clinical data.

5.5 Conclusions from the Simulation Studies

In this chapter, simulated signals were used to evaluate and improve pre-processing, lead reduction and feature extraction methods. Furthermore, the influence of patient specific anatomical variations were analyzed and their impact on the concentrations estimation were assessed. The main conclusions from these studies with regard to an application to clinical data are:

- The bandpass filter bounds minimally influencing the feature values are: 0.1 Hz for the high-pass and 70 Hz for the low-pass filter.
- Maximum amplitude transformation is an adequate choice for lead reduction.
- A patient-specific approach or alternatively a global approach with a patient-specific correction is needed.

These findings were incorporated into the methodology of the next chapter describing the application to clinical data.

Application to Clinical Data

6.1 Introduction

Up to this point of the thesis, all findings were based on simulated data. Nevertheless, keeping an application in a clinical environment in mind, the methods need to be applied to and evaluated on patient data. These data need to consist of simultaneously measured electrocardiograms (ECGs) and measurements of the ionic concentration in the blood. Data from hemodialysis patients meet these needs as an ECG can be easily recorded during a dialysis session and the patient is expected to show concentration variations. An example for the expected concentration values were reported in [67]: pre-dialysis blood potassium concentration ($[K^+]_b$) was 5.23 ± 0.76 mmol/l, post-dialysis $[K^+]_b$ was 3.67 ± 0.25 mmol/l (mean \pm standard deviation). Consequently, hemodialysis patients are a favorable data source for studies on the interplay of ECG morphology and ionic concentration and these kind of data have been widely used in previous investigations. However, blood calcium concentration ($[Ca^{2+}]_b$) variations are usually not as prominent in the dialysis sessions as in the simulation studies presented in this work where extracellular calcium concentration ($[Ca^{2+}]_o$) was varied between 0.6 mmol/l to 3.0 mmol/l. This complicates the evaluation of a $[Ca^{2+}]_b$ estimation method.

In the following, the methods optimized in the last chapter will be applied to a dataset including ECG recordings with simultaneous discrete measurements of ionic concentrations in the blood from dialysis patients. Considering findings from Section 5.4, different patient-specific regression techniques for $[K^+]_b$ estimation will be used. The performances of the methods will be compared to each other and to approaches from literature. The estimation of $[Ca^{2+}]_b$ was not possible due to the small variety of concentration values in the dataset ($[Ca^{2+}]_b$ mean \pm standard deviation: 1.16 mmol/l \pm 0.15 mmol/l).

6.2 Dataset

The data used for this study were taken from two databases (“Calcio Profilato” and “ECG Potassio”) provided by the University of Bologna, Department of Electrical, Electronic, and Information Engineering “Guglielmo Marconi”. During dialysis sessions, the ECGs of 34 patients were recorded with a sampling rate of 1000 Hz. Each patient was monitored in two to four dialysis sessions. All ECGs were acquired with the same ECG device Mortara H12+, Mortara Instrument, Inc., Milwaukee, Wisconsin, U.S.A., however, at different hospitals and in different studies. The ECG recordings lasted for four to six hours. During the ECG recording, the blood ion concentrations were measured discretely five to eight times during a session.

6.3 Methods

6.3.1 Global Model with a Simplified Patient-specific Adjustment

6.3.1.1 Compensation of Unbalanced Datasets

A topic addressed very seldom in other publications dealing with concentration estimation is the correction of unbalanced $[K^+]_b$ in the dataset. According to the description in the publications presented in Table 4.1, none of them introduced such a correction. Nevertheless, this might be useful to train the regression method to cover also the less likely but more important pathological cases. This is the reason why a variable bias compensation was introduced by weighting the least square errors during model fitting. The method is dependent on the histogram of the dataset and will be explained by an example in the following: Figure 6.1 shows a histogram of the dataset. The histogram is approximated by a continuous loglogistic function (red line in Figure 6.1(a)) to then describe the distribution by the corresponding probability density function G . Negating G in combination with a shift by +1 yields $G^- = 1 - G$. Finally, G^- is shifted and normalized (Figure 6.1(b)). The tuning parameter w_r which controls the compensation, influences the normalization. The shifting and normalization step can be expressed by

$$G_{scaled}^- = \frac{G^- - w_r \cdot \min(G^-)}{\max(G^- - w_r \cdot \min(G^-))}. \quad (6.1)$$

6.3.1.2 Regression

The clinical ECG signals were bandpass filtered with a phase-free Butterworth filter of fourth order (passband 0.1 Hz to 70 Hz as proposed in Section 5.2.3). Afterwards, time windows

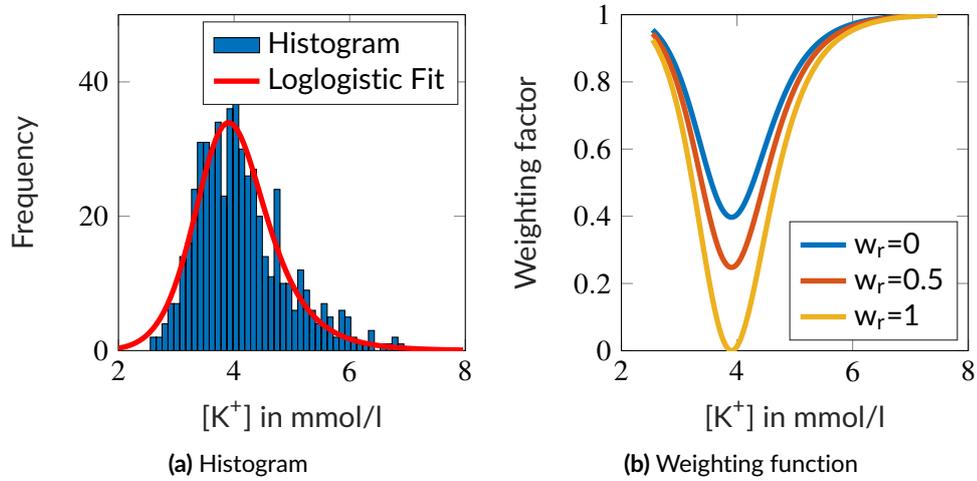


Figure 6.1: (a) Histogram of $[K^+]_b$, the loglogistic fit on it, and (b) the respective weighting functions for different w_r values. The curves are bounded by the minimum and maximum $[K^+]_b$ values in the dataset.

of four minutes were extracted around the time instants when the blood test was performed. For each ECG window, a single beat template was built neglecting artifacts. This pipeline yielded one single beat template per concentration measurement.

The next step was lead reduction. The maximum amplitude transformation was applied to the single beat templates reducing the eight linearly independent leads to one showing a maximized T wave amplitude. This single lead single beat signal was then analyzed with the feature extraction algorithms introduced in Section 5.2.1.

Instead of using the six features as proposed in Section 5.3, the three features showing the maximum absolute value of the Spearman correlation were selected for the mapping from ECG signals to $[K^+]_b$ which were T amplitude, T downslope and T upslope. A regularized weighted least squares problem was solved for a polynomial of third order to fit the model. The weightings were dependent on the distribution of $[K^+]_b$ values in the dataset used for the fitting as described in Section 6.3.1.1. Three different values for w_r were chosen: 0, 0.5, 1. Besides and for comparison, one experiment was performed without weighting. For regularization, first order Tikhonov was used minimizing the L2 norm of the found solution. The regularization parameter was determined by evaluating the curvature of the L-curve as proposed in [98].

The model was finally evaluated with a leave one patient out cross validation, i.e. all patients except one (training set) were used to fit a global model which was then applied to the test patient. Regularization parameter determination and feature selection were performed with only the training data. The simplified patient adjustment of the global model was carried out for the test patient in the following way: the model was applied to the first session of the test patient and the estimation errors for all blood samples in this first session were averaged. This averaged error was subtracted from the estimation with the global model. Thus, a constant over- and underestimation due to patient-specific properties should be prevented.

6.3.2 Patient-Specific Model

In contrast to the methodology described in Section 6.3.1, the patient-specific approach skips the compensation of an unbalanced dataset and the generation of a global model. The idea behind this approach was to build the regression model only with data of one patient. For this, ECG templates were extracted, leads were reduced, and features were calculated as explained in Section 6.3.1.2 with one slight difference: the lead transform was calculated for each template separately. The application of a template-specific lead transform was introduced since there were no inter-patient variations expected and results from previous studies showed that this might be appropriate, too [84].

The first and second session of the patient were used to fit a regularized (the regularization parameter was fixed to 0.01) linear model relying on three features: T amplitude, T upslope and T downslope. Two sessions were used to achieve a more robust solution. In return, this meant that patients with only two sessions (14) had to be omitted.

6.4 Results

6.4.1 Global Model with a Simplified Patient-specific Adjustment

The results of the global model approach for different weightings are shown in Table 6.1. It gets apparent that the regularization was influenced by the choice of w_r : when changing w_r from zero to one, the regularization parameter increased. With the introduction of the weighting, the overall absolute error of the dataset decreased slightly for $w_r = 0$ and $w_r = 0.5$. However, for concentrations larger than 5.5 mmol/l, the mean absolute error improved by up to 0.18 mmol/l whereas the absolute error and its standard deviation for concentrations less than 5.5 mmol/l differed by 0.01 mmol/l only. The weighted standard deviation of the whole dataset accounts for the weighting of the respective samples and increased from 0.77 mmol/l to 1.08 mmol/l.

Table 6.1: Determined regularization parameters and estimation errors ε for the global approach. All unsigned errors are given as mean \pm standard deviation. Errors are given additionally for only hyperkalemic ($\varepsilon_{\text{hyper}}$, $[\text{K}^+]_{\text{b}} > 5.5$ mmol/l) and non-hyperkalemic ($\varepsilon_{\text{nonhyper}}$, $[\text{K}^+]_{\text{b}} \leq 5.5$ mmol/l) samples. Furthermore, the standard deviations of the dataset (Std data) and the weighted standard deviation (Std data weight) are listed. All values except the weighting are given in mmol/l.

Weighting	λ	ε	$\varepsilon_{\text{nonhyper}}$	$\varepsilon_{\text{hyper}}$	Std data weight	Std data
none	[1.58;1.61]	0.48 \pm 0.37	0.45 \pm 0.33	1.03 \pm 0.50	0.77	0.77
$w_r = 0$	[1.45;1.58]	0.47 \pm 0.37	0.44 \pm 0.34	0.92 \pm 0.51	0.90	0.77
$w_r = 0.5$	[1.45;1.58]	0.47 \pm 0.37	0.44 \pm 0.34	0.88 \pm 0.51	0.95	0.77
$w_r = 1$	[1.58;1.81]	0.48 \pm 0.37	0.45 \pm 0.35	0.85 \pm 0.50	1.08	0.77

6.4.2 Patient-Specific Model

The results for the patient-specific solution are given in Table 6.2. As expected, results compared to the globally adjusted model shown in Table 6.1: Although the overall standard deviation of the estimation error was almost the same, the mean absolute error decreased from 0.48 mmol/l to 0.37 mmol/l. Furthermore, the mean absolute errors of the samples showing a concentration above 5.5 mmol/l dropped dramatically from 1.03 mmol/l (no weighting) to 0.60 mmol/l.

Table 6.2: Determined regularization parameters and estimation errors for the patient-specific approach. All unsigned errors are given as mean±standard deviation. Errors are given additionally for only hyperkalemic ($\epsilon_{\text{hyper}}, [\text{K}^+]_{\text{b}} > 5.5 \text{ mmol/l}$) and non-hyperkalemic ($\epsilon_{\text{nonhyper}}, [\text{K}^+]_{\text{b}} \leq 5.5 \text{ mmol/l}$) samples. Furthermore, the standard deviations of the dataset (Std data) and the weighted standard deviation (Std data weight) are listed. All values except the weighting are given in mmol/l.

Weighting	λ	ϵ	$\epsilon_{\text{nonhyper}}$	ϵ_{hyper}	Std data
none	0.01	0.37±0.30	0.34±0.33	0.60±0.40	0.79

6.5 Discussion

6.5.1 Used Data

As already described in the last chapter, one main drawback of this study is the usage of data only from dialysis patients. This could lead to a biased method development and evaluation. Furthermore, the quality of the data varies. Although the pre-processing and template creation should increase the signal to noise ratio (SNR), it was not ensured that this was successful for all signals in the dataset. A manual exclusion of data from patients showing certain pathologies influencing the ECG as in [57] was not performed here. This, however, could further increase the performance of the proposed methods. An estimation of $[\text{Ca}^{2+}]_{\text{b}}$ as described in Section 5.3 could not be performed since data with an adequate concentration variance (standard deviation = 0.15 mmol/l) was not available. Besides the bias in the dataset, problems occurred during the pre-processing of the data. Originally, the dataset consisted of more patients and measurements than described in Section 6.2. However, it was not always possible to add them to the evaluation since blood measurements were not taken during ECG recording, or since a failure of the ECG recording. Furthermore, patients with only one dialysis session had to be excluded. Overall, an optimization of the pre-processing could further improve results.

6.5.2 Regression Model

A regularized polynomial of third order was chosen for the application to patient data in contrast to the neural network as regression model used in Section 5.3. This was mainly due to the fact that regularization and weighting could easily be implemented. First tries with a neural network and a weighted loss function for the training did not deliver results as good as the presented ones. This might be connected with the larger number of parameters for the network, an improper selection of learning rates and activation functions or other hyper-parameters of the network.

6.5.3 Compensation of Unbalanced Datasets

The selection of the histogram fitting function for compensating the bias in the dataset was empiric and not evaluated against other possibilities for bias compensation. Furthermore, the result of the weighting function is dependent on the bin size of the histogram which was fixed to 0.1 mmol/l. Nevertheless, the weighting seemed to be more reasonable than e.g. randomly dropping a part of the most frequent samples to equalize the dataset as the amount of datapoints would have been severely reduced. Besides, the Synthetic Minority Over-sampling Technique (SMOTE) [99], an augmentation strategy, did not deliver acceptable results [100]. The introduction of w_r allowed an increased weighting of the pathological concentrations in the fitting process and therefore led to better results on exactly these data with increasing w_r . However, the user must choose an appropriate w_r depending on the desired model: a model being appropriate for most of the concentrations in the dataset (small w_r) or a model appropriate for the whole spectrum of the data (large w_r).

6.5.4 Comparison of Approaches

In the following, the patient-specific and the global approach will be compared to each other and to two approaches from literature. The work by Attia et al. [57] will be analyzed as a first comparative approach. In their work, the authors propose one patient-specific approach and two different global models:

- (a) A global approach trained with data including sessions from patients in the validation group. The sessions from the training were not included in the validation set.
- (b) A further global approach trained on sessions from those patients that were not included in the validation group.

Since the correction proposed in Section 6.3.1.2 can be seen as an inclusion of the evaluation patient during the training, approach (a) was chosen for comparison.

Common measure for comparison are the mean absolute error and its standard deviation (Table 4.1). However, mean and standard deviation are dependent on the concentration distribution in the dataset [60]. Besides these measures, the authors give selected plots that

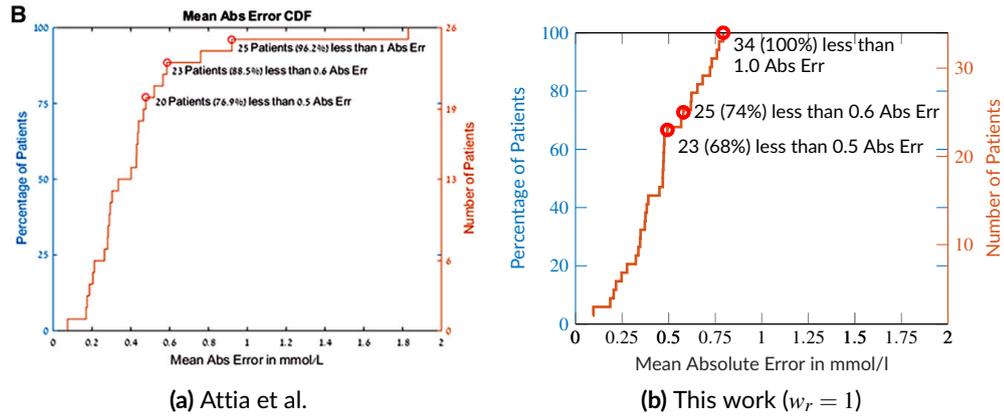


Figure 6.2: Cumulative mean absolute error for the global corrected approaches. (a) Global approach by Attia et al. (b) Global model with patient-specific adjustment from this work. Figure (a) is adapted from [57], licensed under Creative Commons Attribution Non-Commercial License.

were reproduced with the methods suggested in this work. Figures 6.2 and 6.3 visualize the distribution of absolute errors over a cumulative number of patients for the patient-specific and global approaches by Attia et al. and for the respective methods described in this work. In comparison to the percentage of patients showing errors less than 0.5 mmol/l, 0.6 mmol/l and 1 mmol/l, the global adjusted approach proposed in this work outperformed the method by Attia et al. only regarding the maximally observed mean absolute error. When using the correlation plot in Figure 6.4(a) instead of the mean absolute errors, it follows that the method by Attia et al. was not capable of predicting concentrations above 5 mmol/l in their dataset. If concentrations above 5 mmol/l cannot be detected, the method cannot be used for the detection of hyperkalemia ($[K^+]_b > 5.5$ mmol/l). Although the method from this work tended to underestimate higher concentration values, it was still capable of detecting concentrations above 5.5 mmol/l (Figure 6.4b).

The patient-specific approach presented in this work excelled the method by Attia et al. at the upper two estimation error thresholds (Figure 6.3). Both approaches show a good correspondence between estimated and measured concentrations (Figure 6.5). In comparison with the global approaches, the patient-specific method allows a more robust estimation of concentrations above 5 mmol/l. Comparing the patient-specific model results to the one from an adjusted global model, the high discrepancy of the results underlines the need for a more advanced patient adjustment of the global model. However, the global adjusted model found in this work offered two main advantages compared to the patient-specific model: first, only one session from a patient was needed to adjust the model; second, it was possible to fit more complex models with the higher amount of data (polynomial of third order instead of first order).

The work by Corsi et al. [56] will be assessed as a second comparative approach in the following. The authors proposed only a global adjusted model. The results are visualized in form of a Bland-Altman plot in Figure 6.6(a). Again, in order to compare the results better to the ones obtained in this work, the Bland-Altman plots of the global approaches presented in

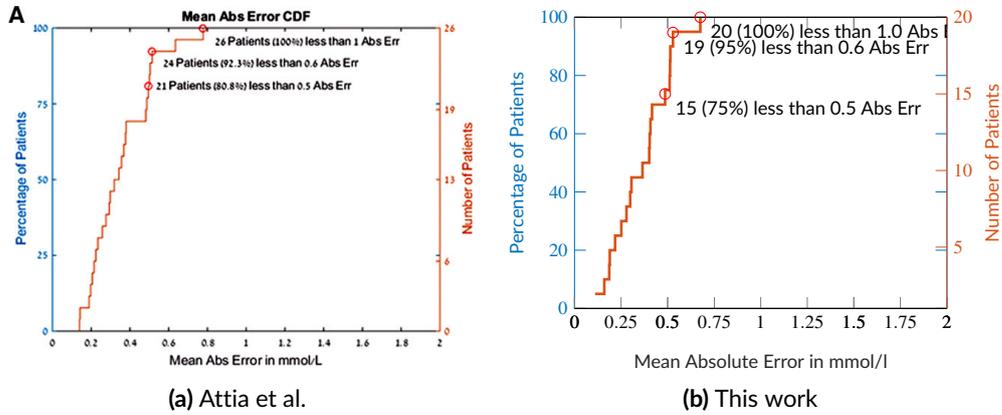


Figure 6.3: Cumulative mean absolute error for the personalized approaches. (a) Personalized approach by Attia et al. (b) Personalized model from this work. Figure (a) is adapted from [57], licensed under Creative Commons Attribution Non-Commercial License.

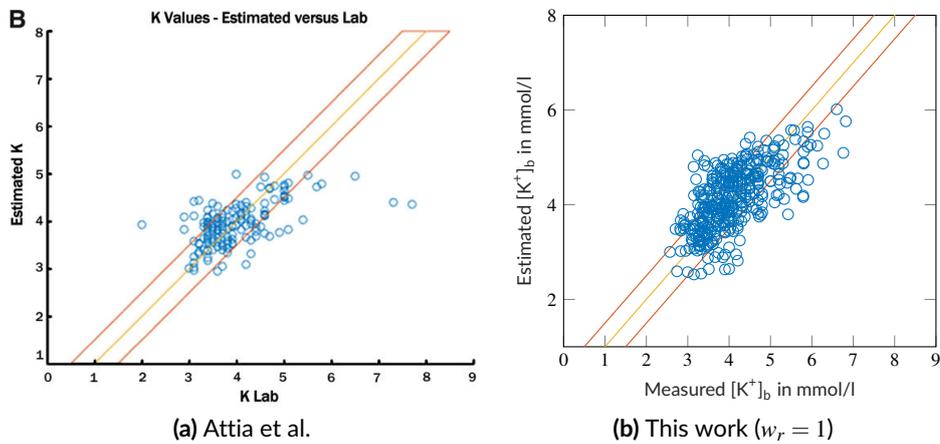


Figure 6.4: Correlation plot of estimated and measured $[K^+]_b$ for the global corrected approaches. The yellow lines indicate the perfect fit between estimation and measured concentrations. Points within the red lines show an error of less than 0.5 mmol/l. In (a), K Lab corresponds to the measured concentrations; all values are given in mmol/l. Figure (a) is adapted from [57], licensed under Creative Commons Attribution Non-Commercial License.

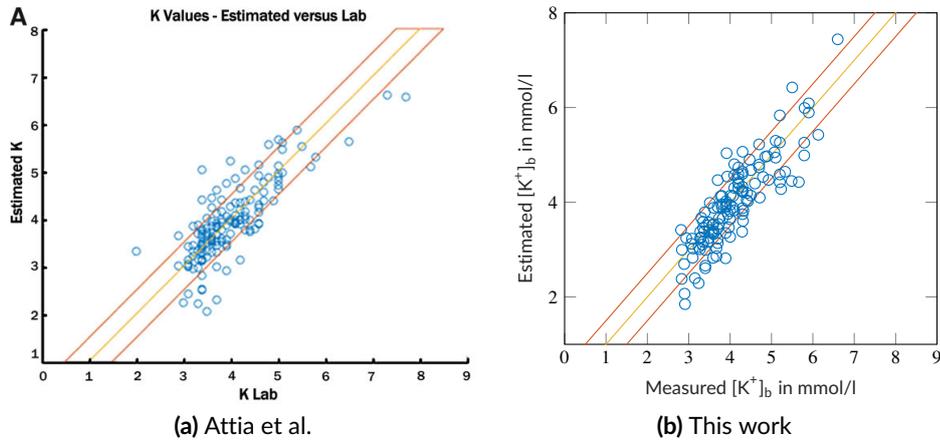


Figure 6.5: Correlation plot of estimated and measured $[K^+]_b$ for the personalized approaches. The yellow lines indicate the perfect fit between estimation and measured concentrations. Points within the red lines show an error of less than 0.5 mmol/l. In (a), K Lab corresponds to the measured concentrations; all values are given in mmol/l. Figure (a) is adapted from [57], licensed under Creative Commons Attribution Non-Commercial License.

this work were created (Figure 6.6(b)-(c)). Similar to the global approaches presented in this work, the method by Corsi et al. tends to underestimate higher $[K^+]_b$ and overestimate lower $[K^+]_b$ (Figure 6.6(a)-(c)). The patient-specific approach presented in this work does not show this behavior (Figure 6.6(d)). By comparing Figure 6.6(b) and (c), the effect of the weighting gets apparent: higher concentrations are underestimated more seldom. In return, the errors in the physiological and most frequently measured range (around 4 mmol/l) increased.

6.5.5 Amplitude Dependency of Features

An important finding from this study was the importance of amplitude and amplitude dependent features in the fitting process. All three features, T wave amplitude, upslope, and downslope are dependent on the amplitude. Although the beat templates were normalized to an R peak amplitude of one, the dependence on the T wave features from the T wave amplitude remains. To visualize this, a lognormal function was used to represent a T wave similar to the experiments in Section 5.2.1. The features T downslope, T upslope, T downslope divided by the square root of T amplitude (T_s/\sqrt{a} , as proposed in [57]) and T downslope divided by T amplitude (T_s/a , as proposed in [56]) were calculated for different amplitudes of the synthetic T wave. All features show a dependence on the amplitude which might be problematic. The origin of an amplitude change was already hypothesized to be linked to a change in the extracellular liquid during dialysis as discussed in [101] and in [102]. If this is true, the result of the concentration estimation methods might be partly or fully dependent on the extracellular liquid change. However, a final proof of the hypothesis from [101] and [102] is still missing. This is directly connected to the problem that the studies by Corsi et al. [56] and by Attia et al. [57] as well as this study utilized only ECGs

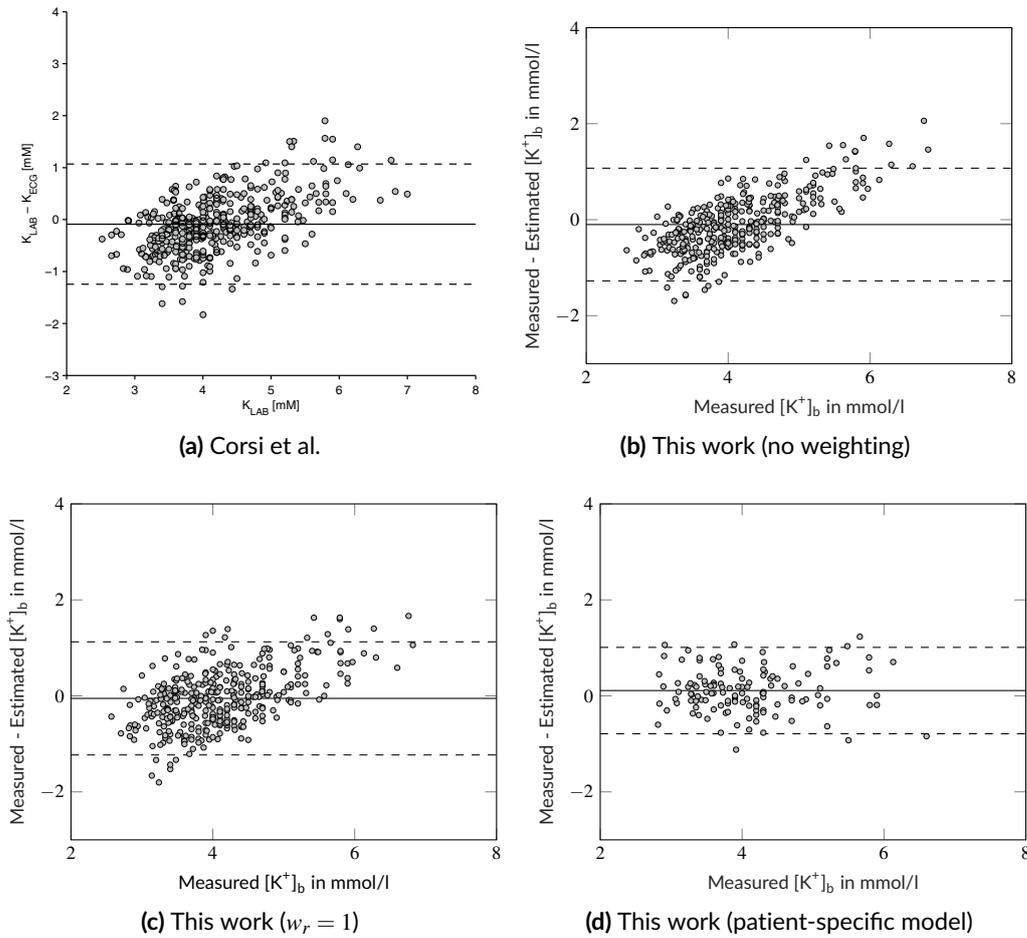
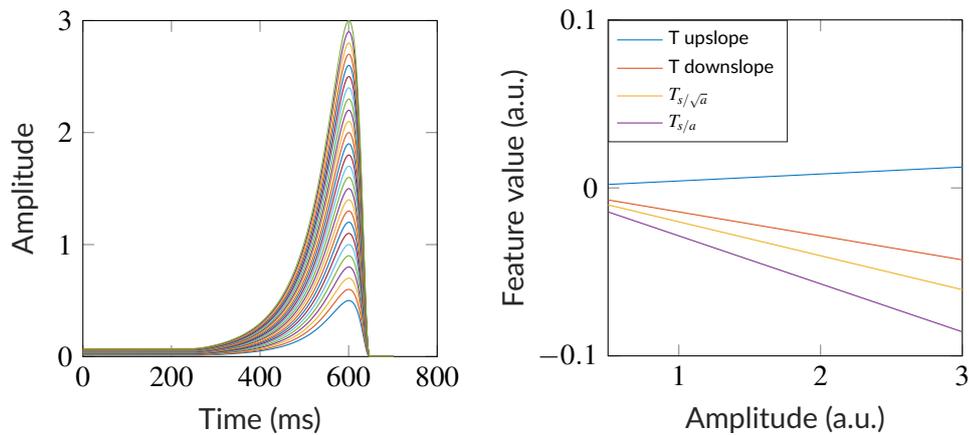


Figure 6.6: Bland-Altman plots for the approach by Corsi et al. and the ones presented in this work. Figure (a) is adapted from [56], licensed under Creative Commons Attribution Non-Commercial License.

from dialysis patients, which in general can cause a biased evaluation. However this choice of the dataset allowed a controlled and reliable data acquisition as already explained in Section 6.1.

6.5.6 Differential Diagnosis

Corsi et al. showed that their proposed feature $T_{s/a}$ for concentration estimation could also be influenced by changes in patients suffering from the Long-QT2 syndrome [56]. This pathology influences the conductivity of certain potassium channels. The finding by Corsi et al. was substantiated in a study confirming that the proposed features might also be dependent on the pathological block of a potassium channel [103]. However, data availability was limited and a final statement could not be made. The work by Corsi et al. suggests that the Long-QT2 syndrome might only lead to a constant offset of the estimation result. Consequently, a fixed compensation or the application of a patient-specific model might help



(a) Lognormal functions with different amplitudes. (b) Features including slopes and amplitudes.

Figure 6.7: Experiment with different lognormal curves shown in (a). Four features were extracted. All are dependent on the amplitude.

to overcome this problem. Other diseases could influence further T wave features in a very similar way. These diseases have to be identified and the changes have to be compensated in an estimation method.

6.5.7 Errors in Ground Truth Measurements

Blood tests are the common state-of-the-art method for ion concentration measurement. These were used in this study as ground truth method. In a recent study [104], Friedman et al. investigated the variability of blood tests. In 1170 fasting and resting patients, potassium concentrations were measured repeatedly in a time frame of 60 min every 30 min and results were compared. Approximately 12% of the repeated measurements showed a difference of 0.5 mmol/l. 20% of the patients had at least one difference exceeding 0.5 mmol/l. The maximum deviation was 2.5 mmol/l. The calculated mean deviations were in the range of 0.26 mmol/l with a standard deviation of 0.25 mmol/l. These mismatches could lead to pseudo-normokalemia in hyperkalemic patients as well as to a wrongly diagnosed hyperkalemia in normokalemic patients. Such variations could be caused either by application errors as well as technical shortcomings, or by a physiological variability in the patients. In spite of everything, these measurement deviations can lead to a misfitting of the potassium estimation model and to biased error values.

Table 6.3: Results of hyperkalemia classification approaches from literature and two naive threshold-based methods from this work. The approaches by Galloway et al. and Lin et al. are based on convolutional neural networks (CNNs). PPV: Positive Predictive Value; NPV: Negative Predictive Value. The table is adapted from [60].

Work	Accuracy	Sensitivity	Specificity	PPV	NPV
Wu et al.[105]	62.5	60	65	n/a	n/a
Tzeng et al.[106]	n/a	85	79	n/a	n/a
Galloway et al.[107]	76.1-80.4	78.1-80.5	75.2-81.3	13.8-18.1	97.6-98.5
Lin et al.[59]	n/a	50.8	96.0	26.9	98.5
Global ($w_r = 1$)	95.8	31.8	100	100	95.7
Pat-spec.	96.1	55.6	99.1	83.3	96.8

6.6 Summary and Outlook

In this chapter, two approaches for potassium concentration estimation were presented. Methods for an optimized pre-processing were selected based on the findings from the simulation studies in Chapter 5. The proposed methods were comparable to or better than methods from literature dependent on the performance parameter. For the first time, the distribution of the dataset was considered in a global model during the fit of a polynomial regression leading to a more precise estimation of concentrations above 5.5 mmol/l.

Apart from these results, there is room for further work: If only the diagnosis hyper- and hypokalemia is of interest, a classification approach can be used as already done in [105, 106] or more recently using a deep learning strategy in [59, 107]. Table 6.3 gives an overview on the performance of methods for hyperkalemia detection from literature. In addition, the methods proposed in this work were used as classifiers by applying a threshold at 5.5 mmol/l on the regression result and performance parameters were calculated. Especially the patient-specific approach performs already comparably to the methods from literature although no optimized technique for classification was applied. A comprehensive investigation on this topic could lead to a robust hyperkalemia detection.

Besides the use of a classification approach, the detection of unreliable templates and feature values could be implemented as in [57] with a Kalman filter. This could decrease estimation errors of the proposed methods.

Compared to the evaluation in Section 5.3, an estimation of $[Ca^{2+}]_b$ was not performed. This was due to the lack of data with a representative variety of $[Ca^{2+}]_b$ values. However, if data were accessible, the methods described here should be applied for $[Ca^{2+}]_b$ estimation, too.

Last, an additional evaluation and improvement of the patient-specific adjustment of the global model could be beneficial for a better estimation. The advantage of a global model being fitted with many data allows for a more complex model than in the single-patient approach. An alternative to adjusting the global model as done in the presented global approach with the subtraction of a patient-specific error, could be the adjustment of the inputs. As shown in Section 5.4, features are dependent on patient-specific anatomical properties. These could be estimated and features could be “normalized” using measures quantifying the anatomical properties.

As the results in Table 6.3 suggest, a deep learning approach for the patient-specific adjustment of the signals, for the adjustment of the global model, or for a direct regression could be beneficial. Patient-specific properties can influence the ECG as shown in Section 5.4, hence, they influence the features and make the estimation based on a global model more difficult. The only possibility is an advanced patient-specific adjustment which can be delivered e.g. by a deep learning approach.

Conclusion

In this part of the thesis, a complete pipeline for the estimation of ionic concentrations was presented. Simulations were used for the optimization of feature extraction methods, pre-processing steps, and to estimate possible influences of patient-specific properties that need to be addressed when designing the concentration estimation method. The results from the simulation studies were incorporated into the two proposed estimation methods, namely a fully patient-specific model and on a global model with a simplified patient adjustment. In the global model, the distribution of the dataset was considered during the fit of a polynomial regression leading to better results for the pathological cases. Nevertheless, if the application allows for a patient-specific model, this should be preferred due to more precise concentration estimations. This could be imaginable for example for the application with dialysis patients visiting a clinical site regularly.

Together with the studies already presented in literature, this work not only proofed the feasibility of an ECG-based concentration estimation but also showed that a clinical application becomes more and more realistic. This will deliver an additional benefit to researchers helping them to better investigate the connection between concentration shifts and sudden cardiac death (SCD). Moreover, patients suffering from concentration shifts will benefit from a more rapid diagnosis of ion concentration imbalances and therefore an improved outcome.

PART III

LOCALIZATION OF ECTOPIC FOCI WITH THE ECG

Introduction to the Topic

Ventricular ectopic beats are triggered by ventricular cells outside the region of pacemaker cells. These beats are considered as normal as long as their frequency is limited. However, they can also cause ventricular tachycardia and this can lead to death. In these pathological cases, catheter ablation can be a suitable treatment. During this, physicians insert a catheter into the ventricles, try to find the triggering source and ablate, i.e. destroy the cells by extreme heat or extreme cold. However, manually locating these spots using sequential catheter mapping can be very time consuming. This is why clinicians have developed manual rule-based methods for identifying these spots based on electrocardiogram (ECG) interpretation. They usually rely on QRS morphologies in the single leads and an algorithm to map these on a region of the heart where the excitation origin is probably located [108, 109]. A possible automatic solution is the utilization of ECG imaging (ECGI) where body surface potentials (BSPs) are used to calculate the sources, e.g. transmembrane voltages (TMVs), on the heart [110]. With the TMV time courses, the activation pattern can be determined and the spatial source of the activation can be detected. This information can consequently be used during the ablation treatment to directly navigate to the location of the ectopic region. However, a major drawback of ECGI is the need of a patient-specific geometry and the need to solve a highly underdetermined problem as the number of measurements on the body surface usually goes below the number of sources in the heart. To overcome these drawbacks, there have been ideas to introduce ventricular segments and use the ECG information for a classification of the original segment. This was for example presented by Kaiyue et al. in [111]. They mapped the information from the ECG to 11 regions on the ventricles using support vector regression (SVR), random forests, gradient boosting decision tree and Gaussian naive Bayes classifiers. Similarly, Yang et al. utilized a convolutional neural network (CNN) for the localization of segments on the ventricular surface containing the origin of the excitation [112, 113]. However, in the last study, the authors still used a patient geometry.

Knowing these studies, the question arises if there is an automatic method to directly estimate the excitation origin from the BSPs without an ECGI or classification step or patient geometries but directly outputs one set of coordinates describing the anatomic position within the heart. One main advantage might be that the direct solution of the problem might be more

robust than methods with various intermediate steps each being prone to errors. However, the effectiveness of this idea must be proven. Subsequently, a method for a clinical application must be designed. These two steps were the aim of this project. The general applicability was evaluated in a proof-of-concept study in Chapter 9. A possible implementation heading to a clinical application is proposed in Chapter 10.

Proof of Concept Study

9.1 Motivation

As explained before, it is not clear if the idea of directly calculating the excitation origin coordinates is practicable and reliable. Therefore, the problem of estimating the excitation origin will be considered as a regression problem. If there is a connection between the body surface potential map (BSPM) and the excitation origin, a regression technique should be able to estimate the coordinates of the excitation origin only with the information from the BSPMs. Additionally, the result from regression was compared to a method utilizing ECG imaging (ECGI) used as reference. To focus on just the evaluation of the feasibility, influences of other variations like patient-specific properties were kept in the following as low as possible. *This study was done in collaboration with Christian Ritter. The results were published in [80].*

9.2 Simulations

Data from a previous study were utilized [114]. Here, 600 simulations with different excitation origins were performed. The de- and repolarization waves were simulated with a cellular automaton on a ventricular mesh ($n = 3340$ nodes, spatial resolution 4 mm). The starting node for each excitation origin was chosen randomly (Figure 9.1(a)-(c)). Using a heterogeneous torso model and 120 defined electrode positions (Figure 9.1(d)-(e)), a 120-lead BSPM was extracted performing a forward calculation.

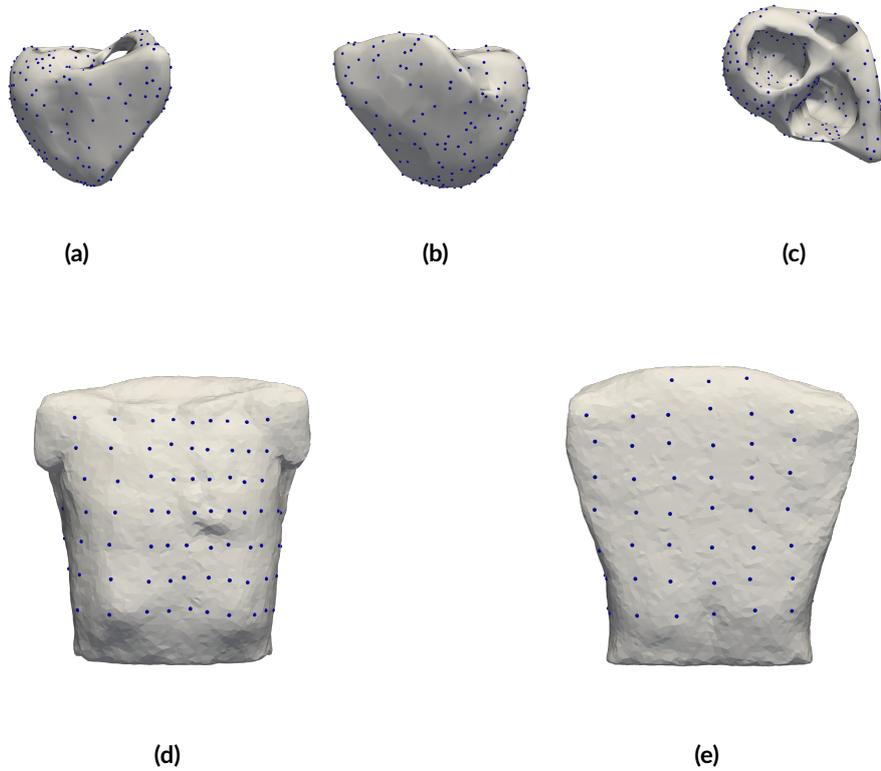


Figure 9.1: (a)-(c) Different views on the ventricular geometry and the excitation origins (blue dots). (d)-(e) Torso model with electrode positions (blue dots).

9.3 Methods

9.3.1 Tikhonov Regularization

To find the excitation origin on the heart, one classical approach from ECGI used here as reference method is to reconstruct the transmembrane voltages (TMVs) on every node on the heart and then determine the first activated node with the calculated TMV courses. The TMVs \mathbf{x} can be found by solving an underdetermined set of equations utilizing 0th order Tikhonov regularization

$$\mathbf{x} = \underset{\mathbf{x}}{\operatorname{argmin}}(\|\mathbf{b} - \mathbf{A}\mathbf{x}\|_2^2 + \lambda^2\|\mathbf{x}\|_2^2). \quad (9.1)$$

with λ^2 being the squared regularization parameter, \mathbf{b} the measured BSPMs and \mathbf{A} the matrix describing the mapping from TMVs to the BSPM which is known from the simulation. Due to the significant influence of the λ on the solution \mathbf{x} , a grid search was applied varying λ from 10^{-8} to 10^{-2} . The λ yielding the minimal Euclidean distance between estimation and

truth was selected for calculating the TMVs.

As already described, the origin of the activation can be found by analyzing the reconstructed TMV signals. In this work, the signals were evaluated 0.01 s after the known start of the depolarization wave. The 5%-quantile of all TMVs was set as threshold and the center of mass of those exceeding the threshold was selected as origin as similarly proposed in [115]. The heart node being closest (minimal Euclidean distance) to the detected origin was used for error calculation. The single coordinates x , y , and z were evaluated separately, i.e. one optimal regularization parameter per coordinate was allowed.

9.3.2 Support Vector Regression

Support vector regression (SVR) was utilized for estimating the global Cartesian coordinates of the excitation origin directly from the surface ECGs. The first 10 principal component scores (showing the maximum variance) were used as input values instead of using all the signals from the 120-lead BSPM. As the original formulation of the SVR allows only one output, three SVRs (one per coordinate) were trained. For the calculation of the errors, six-fold cross validation was used. The method was trained with 500 of 600 samples and evaluated with the remaining 100 samples. The hyperparameters C , ε and σ were determined with the help of a grid search. The parameters were varied between 10^{-5} and 10^5 and the solution with the lowest mean error was chosen for evaluation. As already with the regularization parameter in the Tikhonov regularization, a coordinate specific set of optimal hyperparameters was allowed.

9.4 Results

Results from the two evaluated methods are summarized in Table 9.1. The selected hyperparameters respectively the regularization parameter choice are presented per coordinate. Furthermore, the mean Euclidean distance between estimated and true excitation origin (error) in every direction is given. For Tikhonov regularization, the error was between 5.27 mm and 6.16 mm. The Euclidean distance $d_{global} = \sqrt{d_x^2 + d_y^2 + d_z^2}$ was $d_{global,Tikh} = 10.03$ mm. By contrast, errors obtained with SVR were approximately five times smaller (0.92 mm to 1.08 mm) yielding an Euclidean distance to the true origin of $d_{global,SVR} = 1.73$ mm which is below the mesh resolution of 4 mm. The spatial distribution of the errors are shown in Figure 9.2 for SVR and in Figure 9.3 for Tikhonov regularization. The latter shows the largest errors in the region of the septum on which the BSPMs to source relationship is known to be ambiguous [116]. SVR shows the largest errors in the region of the apex.

Table 9.1: Estimation errors of VEB origin obtained with SVR. Optimal parameters and estimation errors are given for each coordinate separately.

Parameter/ Estimation error	X coordinate	Y coordinate	Z coordinate
$\log_{10}(\lambda)$	-2.5	-3.5	-3
d (mm)	6.16	5.90	5.27
$\log_{10}(\epsilon)$	0.65	-0.74	-10.63
$\log_{10}(C)$	5.02	5.04	4.40
$\log_{10}(\sigma)$	2.18	2.19	1.95
d (mm)	0.92	1.08	0.99

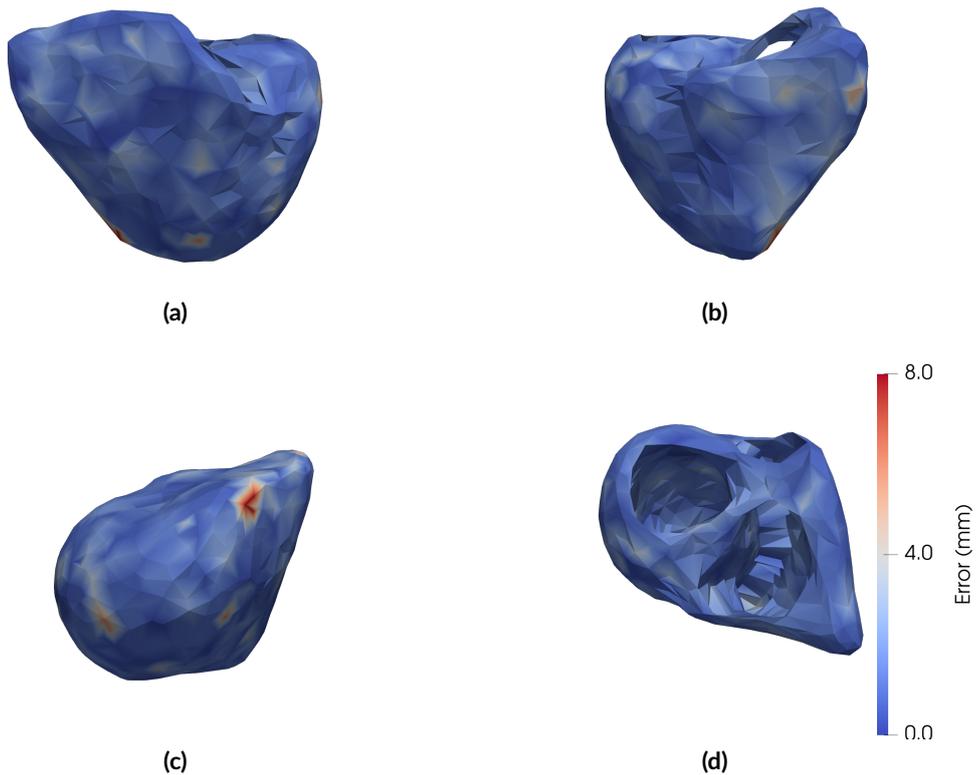


Figure 9.2: Error values for the SVR method visualized on the ventricular geometry from different perspectives.

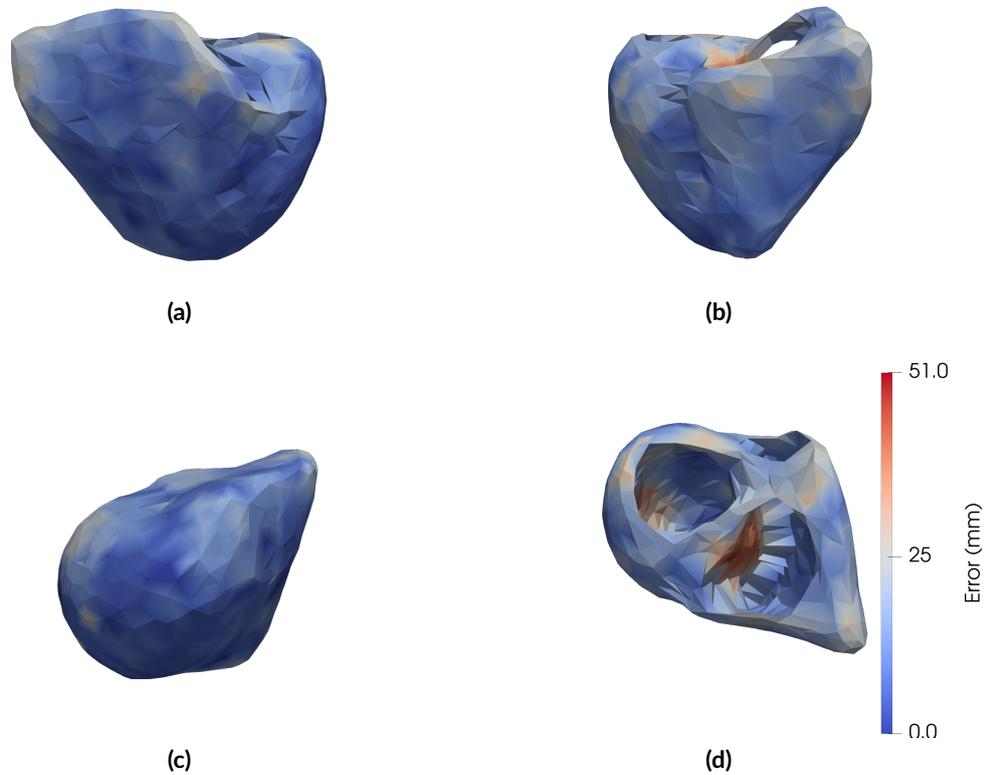


Figure 9.3: Error values for the Tikhonov regularization method visualized on the ventricular geometry from different perspectives.

9.5 Discussion

In this study, 600 BSPMs with different excitation origins were used to estimate the excitation origin. Simulations were all performed with the same patient geometry and the same simple simulation method. On the one hand, this allowed the exclusion of patient-specific or simulation dependent variations on the result, on the other hand, this does not reflect clinical reality. As a consequence, the method as presented above might not meet clinical needs, since a method being applicable to all patients should be generally preferred. Nevertheless, this was not the aim of this proof-of-concept study. Especially for Tikhonov regularization, the mapping matrix \underline{A} was already known from simulation which cannot be granted without a segmented patient geometry from magnet resonance tomography. In contrast, SVR is not dependent on this information. However, the current implementation hardly allows a patient independent application as coordinates were not normalized in any kind. Furthermore, it can be hypothesized that SVR could have learned some kind of mapping matrix \underline{A} from the data.

Moreover, the dimensionality of the BSPMs was reduced arbitrarily to the first 10 scores of the principal component analysis (PCA) showing the largest eigenvalues.

This work relied apart from the fixed patient properties on other optimal preconditions: noise-free signals, optimized hyperparameters, and consequently probable overfitting resulted in good results that are unlikely to achieve in clinical practice without further pre-processing or adjustment steps. These simplifications are usually referred to as “inverse crime”. This holds of course for both methods. Nevertheless, all these limitations do not contradict the general suitability of a regression technique like SVR for the estimation of the excitation origin.

9.6 Conclusion

SVR was able to deliver apart from all limitations an excitation origin estimation error of 1.73 mm which is below the mesh resolution. This promotes the hypothesis that there is a functional connection between the BSPM and the excitation origin that can be exploited with an adequate regression technique to achieve a highly accurate excitation origin estimation. In contrast to methods relying on classical ECGI, the regression technique works without using a patient geometry which might be beneficial in a clinical application.

9.7 Outlook

The main drawback of this proof-of-concept study is the missing variation of patient geometries and conduction properties. In a follow-up study, patient data could be varied with the help of a ventricular shape model as presented in [90]. Furthermore, a variation of the position of the ventricles in the torso, of the conduction velocities and an anisotropy of the velocities could be introduced. These data could then be used with a method estimating all coordinates at the same time using the full BSPM. The additional use of universal ventricular coordinates [91] could introduce a patient-independent coordinate system. Most of the drawbacks were addressed in the project presented in the next chapter 10.

Deep Learning Approach

10.1 Introduction

The proof-of-concept study reported in the last chapter confirmed the hypothesis of a feasible direct estimation of the excitation origin. Nevertheless, the drawbacks of the study comprised limited variations in the simulations. To account for this point, in the following study, a database including different excitation origins, different ventricular geometries with different orientations in the torso and different conduction properties was generated. These data were used to train a fuzzy classifier unifying regression and classification for the estimation of the excitation origin. By doing so, ambiguous solutions could be visualized. *The results in this project were obtained in close collaboration with Steffen Schuler. The findings are based on the the Master's theses of Gerald Moik [89] and Maike Rees.*

10.2 Simulations

10.2.1 Geometrical Variations and Conduction Properties

Ventricular geometries were created using a ventricular shape model presented in [90, 117] available on github [118]. It was generated based on more than 1000 ventricular geometries and includes a mean shape and 100 principal components with their respective variance covering 99.9% of all variation in the dataset. This shape model was extended to output not only a right ventricular endocardial but also a right ventricular epicardial wall with a transmural volume in between. As new points were added to the mesh which have no principal component and variance assigned, they were inter-/extrapolated using a Laplace interpolation based on the existing points. For the left ventricle, there was already a endo- and epicardial wall with a transmural volume in the original mean shape. 1000 instances of the shape model were generated adding the weighted sum of all principal components (three examples are shown in Figure 10.1). The weighting was calculated with a quasi-random

variation (Halton sequence) of the standard deviation of the respective principal component. This weighting was constrained to \pm three times the standard deviation of the respective principal component. Ventricular coordinates were calculated for all 1000 instances of the shape model similar to the method proposed by Bayer et al. [91] These coordinates content for every point an apicobasal, rotational, transventricular and transmural coordinate as shown in Figure 10.3. As the rotational coordinate has a jump between 0 and 1 (2π in the non-normalized formulation), it can be transformed with a sine and cosine function obtaining a continuous course and preserving the uniqueness of the coordinate. Although this adds another coordinate, this facilitates the error calculation e.g. during the training of the regression method. In this study, transmural and transventricular coordinates were treated as binary variables by applying a threshold at 0.5. Next, ventricular fibers were calculated for all the shape model instances using an extended and corrected version of the approach from [87]. Fiber angles were between 40° to 80° for the endocardial surface and -80° to -40° for the epicardial surface. The ventricular geometries were placed into a torso geometry and aligned while varying the roll, pitch and yaw angles as described in Section 5.4. As well, the geometries were translated. For each geometry, three different angle-translation combinations were used as shown in Figure 10.2. Obtaining all these variations, a total number of 3000 geometries were generated.

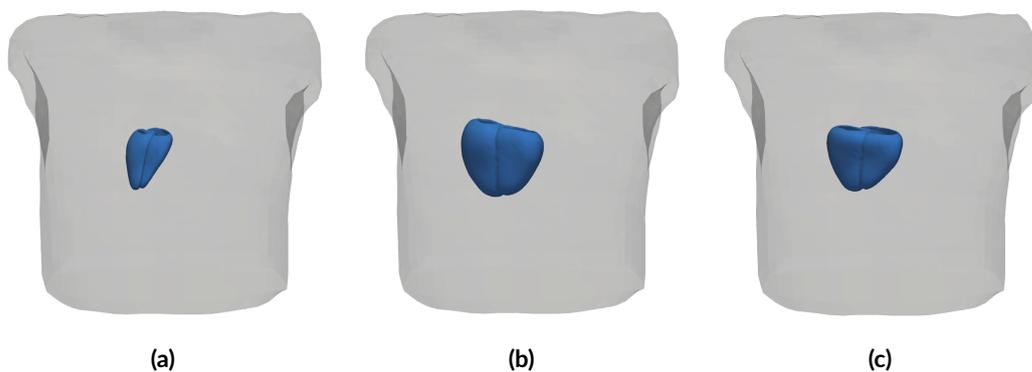


Figure 10.1: Three different shape model realizations. The heart was neither aligned with the torso nor rotated or translated.

10.2.2 Excitation Simulations and Forward Calculation

For each of the 3000 geometries, 560 to 615 (mean 588.802) triggering points were distributed over the left and right ventricular endocardium as well as the epicardium dependent on the area proportion of the respective parts on the sum of all areas. Activation time distribution for all trigger points on all geometries were calculated using the Eikonal approach (FIM Eikonal [119]). The anisotropy ratio was set to 2.7. A change of conduction velocities

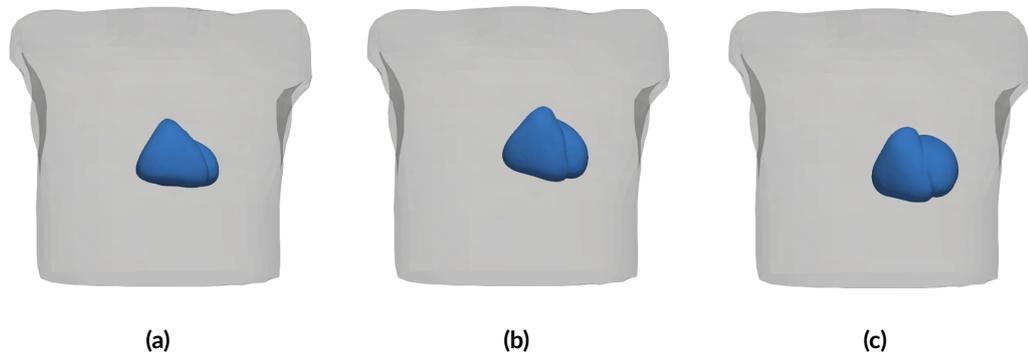


Figure 10.2: Three different positions of the geometry from Figure 10.1(b). The heart was aligned with the torso, rotated and translated.

was introduced by scaling the calculated activation times achieving a conduction velocity between 0.4 m/s and 1.2 m/s. The change was quasi-randomly (Halton sequence) applied to the simulations. Body surface potential maps (BSPMs) were extracted using the transfer matrix from a boundary element method and 200 electrodes on the torso front and back (Figure 10.4). To calculate the surface potentials, a transmembrane voltage (TMV) template was shifted according to the activation times obtained by the Eikonal approach and forward calculated with the transfer matrix. The sampling rate was 1000 Hz. In total, $3000 \cdot 588.802 = 1,766,406$ single beat BSPMs were created with a known excitation origin that can be described by ventricular coordinates. The single beat realizations were padded at the beginning randomly between 0 and 100 ms and randomly cut between 0 and 100 ms after the end of the activation. This was done to simulate an unaligned signal snippet which are usually solely available in the clinical environment. To simulate realistic input data, the BSPMs were overlain with realistic noise from the MIT-BIH Noise Stress Test Database [78, 120]. Baseline wander, muscle artifacts and white Gaussian noise was added to two randomly selected but distantly located electrodes. The noise for all other electrodes was calculated using a spatial Laplacian interpolation of the respective noise types. The noise was amplitude scaled yielding the following average signal to noise ratios (SNRs) over all electrodes per noise type: Baseline wander showed an SNR of 10 dB, muscle artifacts 30 dB, and white Gaussian noise 30 dB, respectively. The single leads of the BSPM with a random weighting between 0 and 1 to obtain different SNRs in different leads. Afterwards, the lead signals were filtered with a Butterworth bandpass filter between 0.5 Hz and 150 Hz and resampled to 500 Hz. Each lead was normalized to the maximum absolute amplitude in the respective lead. The length of all lead signals was 350 samples. The missing 24 electrode signals in the region of the arms were spatially interpolated to obtain a symmetrical shape.

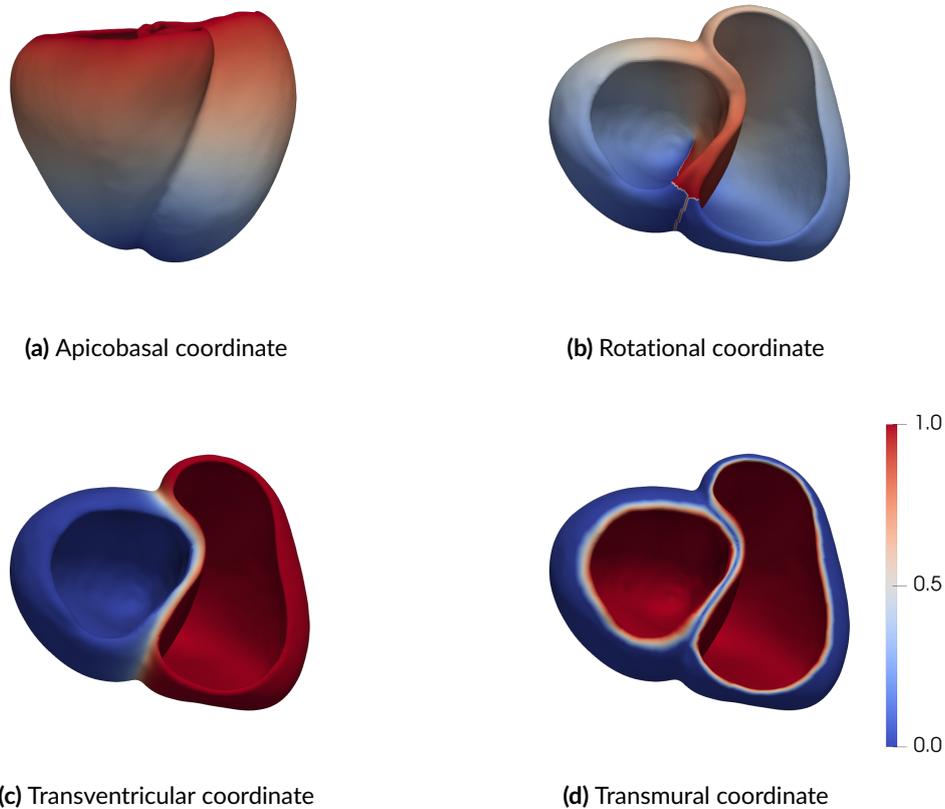


Figure 10.3: The four ventricular coordinates describing each position on and in the ventricles. All coordinates are normalized between 0 and 1.

10.2.3 Generation of Ventricular Classes

In order to formulate the localization task, which is originally a regression task, as a fuzzy classification task, the ventricular surface was divided into 303 classes. Therefore, the ventricular mesh (Figure 10.5(a)-(b)) was converted into a coarse triangular surface mesh (Figure 10.5(c)) containing 303 points. Endocardium and epicardium were separated as visible in Figure 10.5(c). The procedure to determine the class weightings for a focus will be described with an example in the following: Figure 10.5(d) shows a focus (red dot) on the coarse triangular mesh. The three vertices of the triangle containing the focus are determined (orange dots). With the help of barycentric coordinates, the position of the focus can be described easily: In this example, the coordinates are one third for each vertex as the focus is in the center of mass of the triangle. These barycentric coordinates (one third for all) are now assigned as class weights to the 303 classes as every vertex represents one class. All other points (partly visualized with white dots in Figure 10.5(d)) are zero. This procedure is repeated for each focus obtaining always three non-zero values among the 303 classes.

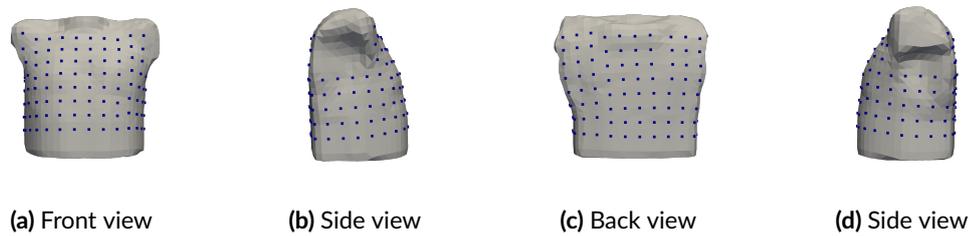


Figure 10.4: Different views on the torso with the 200 electrodes (blue dots). In the region of the upper arms, no electrodes were placed.

Moreover, the number of classes can be reduced by estimating the transmural and transventricular coordinate separately. As the left and right endo- and epicardial triangular meshes always consist of the same amount of points, the classification method can also be trained with only 93 classes out of which 91 (triangles) represented solely the apicobasal and rotational position independent from the transmural and transventricular position. The remaining 2 classes of the 93 were used as binary classes for transventricular and transmural position classification.

10.2.4 Dataset Preparations

The BSPM dataset was splitted into three subsets: a training set (70%, i.e. 700 geometries), a validation set (15%, i.e. 150 geometries) and a test set (15%, i.e. 150 geometries). By dividing the dataset using only the geometries, the scenario of a “new patient” could be simulated.

10.3 Methods

10.3.1 ScaleNet - Activation Onset and Offset Detection

The extracted single beat BSPMs contained ventricular depolarization (QRS complex) and at least parts of the repolarization (T wave). Nevertheless, it is reasonable to only consider ventricular depolarization in the signal as this delivers all the needed information about the origin of the excitation. Reducing the data to only the depolarization delivers a further advantage: The training of a learning method is accelerated since the dimensionality of the problem is reduced and only relevant information is presented to the method. Nevertheless, the start and end of the depolarization in the heart tissue cannot be detected easily only relying on the BSPM. This is the reason why in a first step, a convolutional neural network (CNN) called ScaleNet was trained to detect start and end of the ventricular activation. It should be emphasized that it is not the same as the beginning and end of the QRS

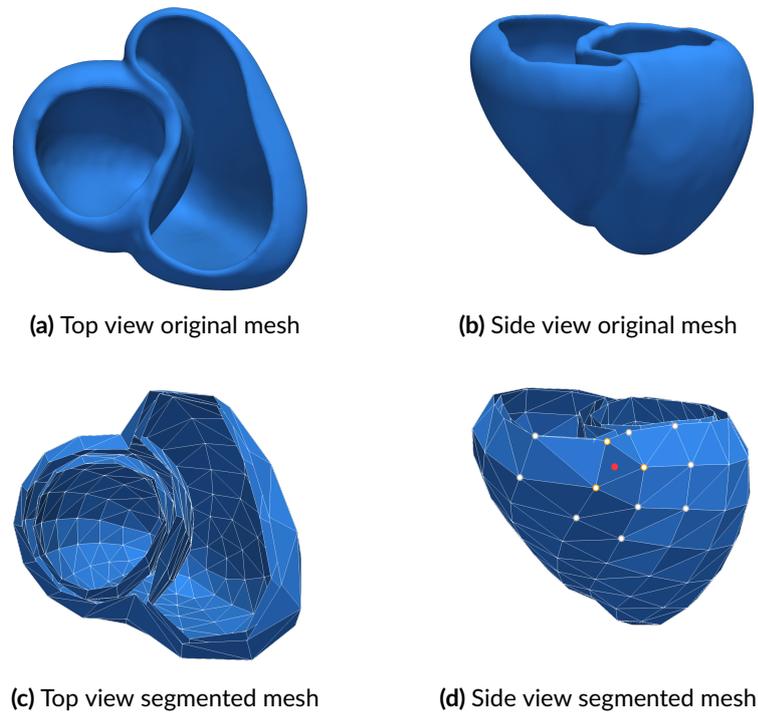


Figure 10.5: (a)-(b) Original mesh from top and side view. (c) Top view on segmented mesh. Endo- and epicardium are separated. (d) Side view on the segmented mesh. An example for an ectopic focus (red dot) is given. The points defining the surrounding triangle are colored in orange. The white dots are examples for the vertices of the triangles not directly surrounding the ectopic focus.

complex. The detection of depolarization start and end delivers a further advantage: in a following localization step, signals can be resampled to a constant length compensating for different conduction velocities and therefore offering a kind of normalized beat facilitating the following part. For the training of this and all following CNNs, the PyTorch framework (v1.7.0) was used [121].

The true start and end of the depolarization was obtained from the Eikonal simulations. For training and evaluation, the single leads of the BSPM were arranged into a two-dimensional matrix with the dimensionality 224×350 as shown in Figure 10.6. The second dimension is time, the other is the electrode signals that were arranged going spirally craniocaudally around the torso starting at the top front.

As architecture, ResNeXt-50 (32x4d) [122] was used. For being applicable to the problem, the input and output layers were changed. The output layer consisted of two neurons (delivering the value for start and end). The input layer was changed to a convolutional layer with a kernel size of five with stride two. The input channel size was set to one, the output channel number was kept at 64. For the training, the following hyperparameters were set:

- Optimizer: ADAM with $\beta_1 = 0.9$ and $\beta_2 = 0.999$, $\gamma = 10^{-8}$
- Batch Size: 680

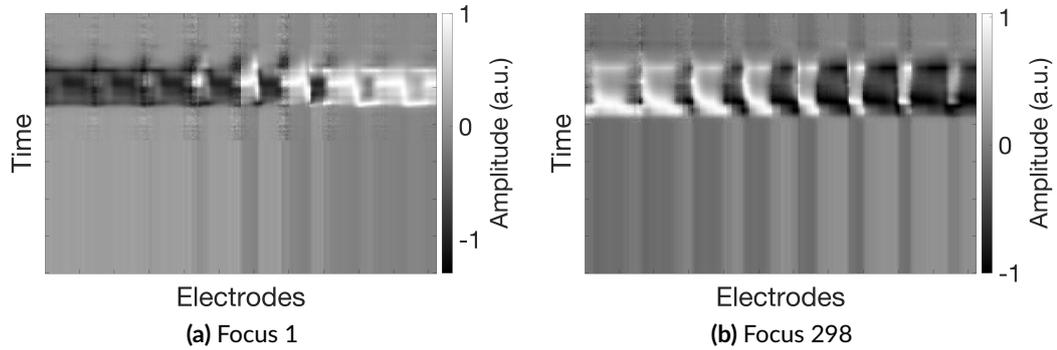


Figure 10.6: Two different BSPM images from different excitation origins. Time and electrodes are visible in the spatial domain. Amplitudes are grey value coded.

- Learning Rate: Cyclic learning rate [123] with a maximum learning rate of $1.623 \cdot 10^{-3}$ and a minimum learning rate of $4.553 \cdot 10^{-4}$; the learning rate is increased for two epochs, and decreased for another two epochs.
- Training time: 90 hours.
- Initialization: Weights and biases of the different layer types were initialized according to the default values of PyTorch.

Details on the structure of the CNN are given in Section A.5 in the appendix.

The averaged mean squared error of both time instants was used as loss function

$$\text{loss}_{\text{scaleNet}} = \frac{\frac{1}{N} \sum_{i=1}^N (l_{s,i} - y_{s,i})^2 + \frac{1}{N} \sum_{i=1}^N (l_{e,i} - y_{e,i})^2}{2}, \quad (10.1)$$

where $y_{s,i}$ ($y_{e,i}$) is the estimated start (end), $l_{s,i}$ ($l_{e,i}$) is the known label for start (end) from one focus simulation and N is the number of evaluated samples.

The minimum and maximum learning rate was obtained from test runs with the training dataset. Therefore, a learning rate sweep was performed for every batch during one epoch varying the learning rate between 10^{-5} and 10^{-2} . The loss (mean squared training error) was calculated for the different learning rates (blue line in Figure 10.7). The result was smoothed in a logarithmic scale (red line in Figure 10.7) obtaining a sigmoid-like shape. With the help of a method similar to the L-curve method presented in [98] and used in Section 6.3.1.2, first the upper bound was found using the maximum of the curvature of the smoothed loss (green line in Figure 10.7). Afterwards, the preceding minimum was determined as lower bound for the learning rate. The applied cyclic learning rate is visualized in Figure 10.8.

10.3.2 LocaNets - Localization of Ectopic Foci

Three further CNNs (LocaNets) were designed: one performing a fuzzy classification with the 303 fuzzy classes introduced in Section 10.2.3 (in the following FuzzyNet303), another

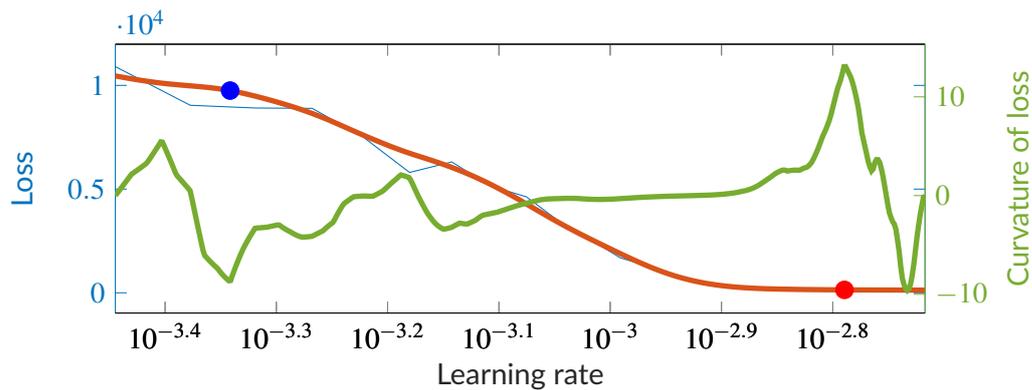


Figure 10.7: Loss for different learning rates in logarithmic scale (blue) and the respective smoothed loss curve (red). Two points were detected as the lower (blue) and upper (red) bound for the cyclic learning rate during training using the curvature of the smoothed loss (green).

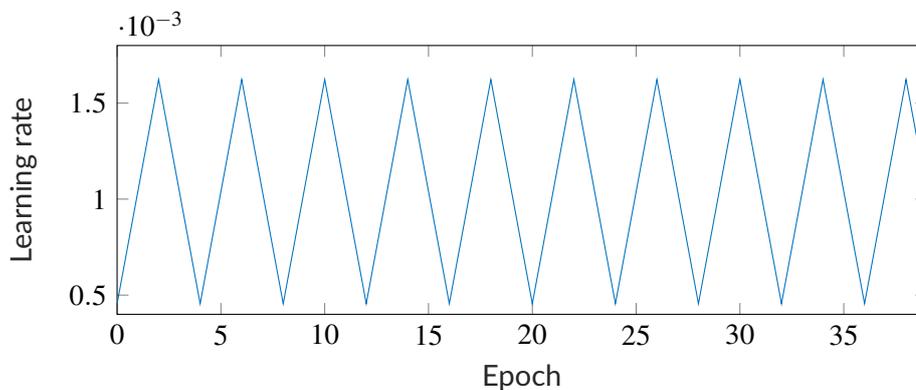


Figure 10.8: Change of the learning rate over the epochs due to the implementation of the cyclic learning rate.

performing the classification of the reduced 91 fuzzy classes (Section 10.2.3) plus two binary classifications for the endo-/epicardium and for the left/right ventricle (in the following FuzzyNet93). The third network directly estimated the ventricular coordinates as regression task (in the following RegressionNet).

All networks used the same input and therefore input layers: With the results of ScaleNet, the signals were cropped and resampled to obtain a constant temporal length of 125. Therefore, the inputs for the networks were two dimensional images of the size 224x125. The output layers were adjusted according to the respective LocaNet. The detailed structures are given in Section A.5 in the appendix.

The loss functions varied between the networks. For RegressionNet, a combination of mean squared error for the rotational (rt_cos and rt_sin) and apicobasal (ab) coordinates, and binary

cross entropy for transventricular and transmural coordinate was used as loss function

$$L_{\text{RegressionNet}} = \text{mse}_{\text{ab}} + \text{mse}_{\text{rt_cos}} + \text{mse}_{\text{rt_sin}} + l_{\text{tm}} \log(\sigma(y_{\text{tm}})) + (1 - l_{\text{tm}}) \log(1 - \sigma(y_{\text{tm}})) \\ + l_{\text{tv}} \log(\sigma(y_{\text{tv}})) + (1 - l_{\text{tv}}) \log(1 - \sigma(y_{\text{tv}})) . \quad (10.2)$$

y denotes the predictions and l the labels. σ is a sigmoid function preventing the argument of the logarithm from becoming zero. The mean squared errors of the non-binary coordinates were defined as

$$\text{mse}_{\text{ab}} = \frac{1}{N} \sum_{n=1}^N (2.5l_{n,\text{ab}} - 2.5y_{n,\text{ab}})^2 , \quad (10.3)$$

$$\text{mse}_{\text{rt_sin}} = \frac{1}{N} \sum_{n=1}^N (\sqrt{l_{n,\text{ab}}}l_{n,\text{rt_sin}} - \sqrt{l_{n,\text{ab}}}y_{n,\text{rt_sin}})^2 , \text{ and} \quad (10.4)$$

$$\text{mse}_{\text{rt_cos}} = \frac{1}{N} \sum_{n=1}^N (\sqrt{l_{n,\text{ab}}}l_{n,\text{rt_cos}} - \sqrt{l_{n,\text{ab}}}y_{n,\text{rt_cos}})^2 . \quad (10.5)$$

The errors of the rotational coordinates were weighted with the labels of the apicobasal coordinate since the rotation is undefined on the apex. As well, a rotational coordinate change is differently scaled in the Euclidean space dependent on the apicobasal position. As there are two rotational coordinates and rotational and apicobasal step size are differently scaled in the Euclidean space, the apicobasal error is weighted with 2.5^2 .

For the fuzzy classification problems, categorical cross entropy was used. The loss function for FuzzyNet303 was

$$L_{\text{FuzzyNet303}} = \sum_{n=1}^{303} l_n \log \left(\frac{e^{y_n}}{\sum_{j=1}^{303} e^{y_j}} \right) , \quad (10.6)$$

where N is the number of classes. For FuzzyNet93, the loss was

$$L_{\text{FuzzyNet93}} = \sum_{n=1}^{93} l_n \log \left(\frac{e^{y_n}}{\sum_{j=1}^{93} e^{y_j}} \right) + l_{\text{tm}} \log(\sigma(y_{\text{tm}})) + (1 - l_{\text{tm}}) \log(1 - \sigma(y_{\text{tm}})) \\ + l_{\text{tv}} \log(\sigma(y_{\text{tv}})) + (1 - l_{\text{tv}}) \log(1 - \sigma(y_{\text{tv}})) . \quad (10.7)$$

This is basically the loss from FuzzyNet303 with an added binary cross entropy loss for the transmural (tm, endo-/epicardium) and transventricular (tv, left/right) classification.

The hyperparameters for training were similar to the ones from ScaleNet except those given in Table 10.1.

To be able to quantify and compare the performance of the approaches, the localization error in millimeter was calculated. For RegressionNet, the ventricular coordinates were mapped back to the Euclidean space and the geodesic distance on the patient-specific heart mesh between label and prediction was computed. For the FuzzyNets, the classification results could be used to identify the segment where the activation started. The predicted fuzzy class values were assigned to the respective vertices visualized in Figure 10.5(d). The triangle with the largest sum of vertex values was identified as the source segment.

Table 10.1: Hyperparameters for the LocaNets that were different from those from ScaleNet. Max/Min LR is the maximum/minimum learning rate used for the application of cyclic learning rate.

Network	Min LR	Max LR	Batch size	Training time
FuzzyNet93	0.0008701	0.007409	1800	48 hours
FuzzyNet303	0.00101	0.006638	1860	48 hours
RegressionNet	0.0004909	0.003229	1800	48 hours

Subsequently, the position of the focus could be localized using the predicted class values (vertex values) as barycentric coordinates (Figure 10.5(d)). Afterwards, the geodesic distance using the individual heart mesh could be calculated as with RegressionNet. In the case of FuzzyNet93, the binary classification results had to be used to determine the transventricular and transmural position of the source triangle before determining the focus position in the triangle. For both FuzzyNets, multiple solutions could be determined. Instead of taking the triangle with the largest sum of vertex values, the second, third, etc. largest could be visualized, too. Hence, it was possible to calculate e.g. the top-two errors.

10.4 Results

10.4.1 ScaleNet

During the training time of 90 hours, ScaleNet was trained on four Nvidia Tesla V100 GPUs for 39 epochs. At epoch 32, the validation error was minimal (Figure 10.9). So, the calculated weights from this epoch were chosen for the evaluation and further steps. The means, medians, standard deviations and interquartile ranges of the detection errors of start and end of the ventricular depolarization are given in Table 10.2 for training, validation, and test dataset. It gets apparent, that the estimation of the end seemed to be more difficult than of the start as standard deviations were larger. However, all parameters stayed below one sample. Figure 10.10 visualizes the errors as boxplot. Most outliers stayed below 10 samples which corresponds to 20 ms. The largest absolute errors were 15.9 samples (31.8 ms) for the start and 16.8 samples (33.6 ms) for the end. Histograms of the errors are additionally given in Figures A.11 and A.12 in the appendix.

Table 10.2: Errors for the detection of activation start and end with ScaleNet are given as mean (m) and standard deviation (std) for the signed (sig) and unsigned (usig) errors. Furthermore, median (med) and the interquartile range (iqr) for the signed errors are given. Training (train.), validation (val.), and test dataset were analyzed separately. All values are in samples. The sampling rate was 500 Hz.

Measure	Train. start	Val. start	Test start	Train end	Val. end	Test end
sig m±std	0.06±0.35	-0.06±0.40	-0.06±0.40	0.03±0.48	0.04±0.68	0.04±0.67
med±iqr	0.06±0.42	-0.06±0.45	-0.06±0.45	0.01±0.54	0.01±0.60	0.01±0.60
usig m±std	0.27±0.23	0.30±0.28	0.29±0.28	0.35±0.32	0.44±0.52	0.44±0.51

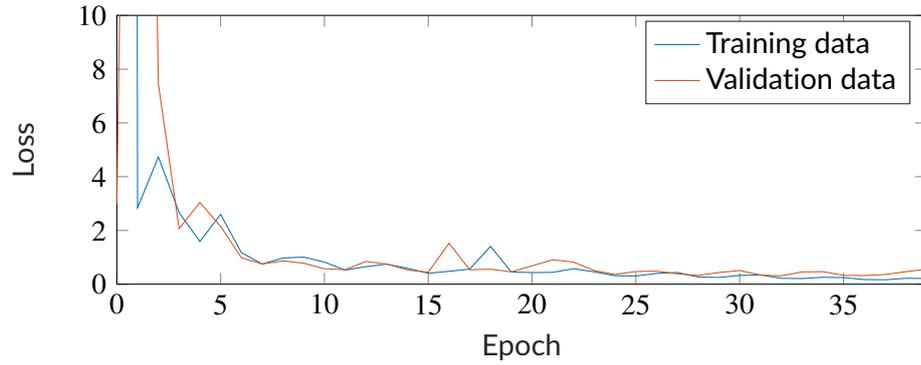


Figure 10.9: Progression of the loss on the training and validation dataset during training. At epoch 32, the validation loss is minimal.

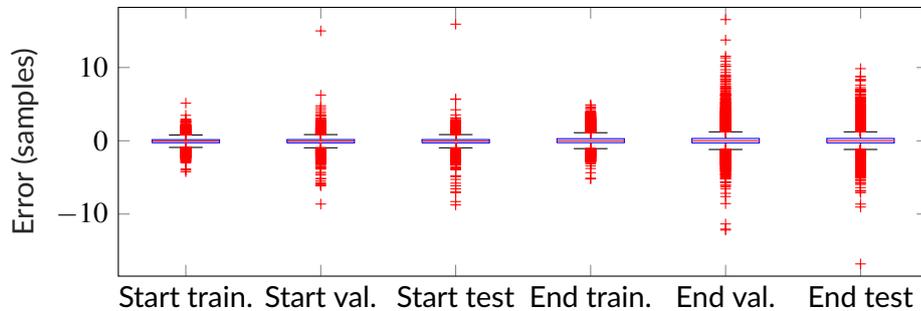


Figure 10.10: Boxplot of the errors of ScaleNet. Start and end detection errors are given for training (train.), validation (val.) and test data.

10.4.2 LocaNets

The networks were all trained for 48 hours, each on four Nvidia Tesla V100 GPUs. This yielded in different total numbers of epochs for the approaches: FuzzyNet303 had 40 epochs, FuzzyNet93 had 44, and RegressionNet was trained for 46 epochs. Hence, computational complexity seemed to increase between the RegressionNet and the FuzzyNets. For evaluation, the epoch with the lowest validation error was considered. This was epoch 15/27/39 for FuzzyNet303/FuzzyNet93/RegressionNet.

Instead of evaluating the used loss values (see Equation (10.2), Equation (10.6), and Equation (10.7)) on the test dataset which are hard to compare, the localization errors of the three approaches are displayed in Table 10.3. FuzzyNet303 performed best regarding all measures, second best was FuzzyNet93 regarding the test localization errors. RegressionNet performed worst. This also reflects in the boxplots of the localization errors (Figure 10.11): RegressionNet showed the largest outliers of all methods in the test dataset. For a more detailed comparison of the error distribution, error histograms are shown in Figures A.13, A.14, and A.15 for all approaches in the appendix. There was no correlation found between

the region (segment) of the focus origin and the error value. As visible in Table 10.4, there was no coordinate with an extraordinary high error compared to the others on the test dataset.

Table 10.3: Localization errors (geodesic distance between prediction and label) as mean (m) and standard deviation (std), and as median (med) and interquartile range (iqr). Training, validation (val.) and test dataset were analyzed separately. All values are in mm.

Method	Measure	Training	Val.	Test
FuzzyNet303	m±std	1.09±0.73	1.67±1.14	1.67±1.11
FuzzyNet93	m±std	1.28±0.82	1.81±1.20	1.82±1.19
RegressionNet	m±std	2.48±1.50	2.59±1.62	2.57±1.56
FuzzyNet303	med±iqr	0.94±0.83	1.52±1.33	1.54±1.34
FuzzyNet93	med±iqr	1.05±0.96	1.65±1.50	1.66±1.51
RegressionNet	med±iqr	2.23±1.83	2.34±1.91	2.33±1.90

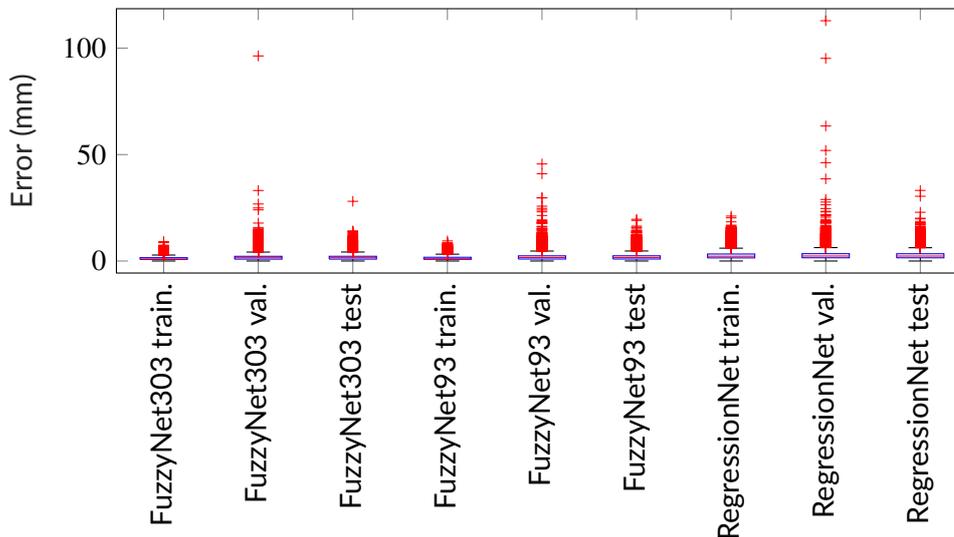


Figure 10.11: Boxplot of the errors (geodesic distance between prediction and label) of the LocaNets for each dataset partition (training (train.), validation (val.) and test).

10.5 Application to Clinical Data

To proof the applicability of the network to patient data, ScaleNet and LocaNets were used on patient data without training on patient data. 67 BSPM recordings (Amycard 01C EP system, EP Solutions SA, Switzerland) from 37 patients with implanted biventricular pacemakers were used to evaluate the methods. Data were already partly utilized in [124] and in [125]. The pacemakers were programmed to pace from the left (epicardial) or right (endocardial) ventricular stimulus electrode. The location of the stimulus electrode was known from a

Table 10.4: Coordinate test errors as mean (m) and standard deviation (std), and as median (med) and interquartile range (iqr) for apicobasal and rotational coordinates. For the binary transventricular and transmural coordinates, the error rate (erate) is given. All values are calculated on the test set and are given in percent.

Coordinate	Measure	FuzzyNet303	FuzzyNet93	RegressionNet
apicobasal	m±std	0.01±1.35	-0.03±1.44	0.09 ±1.97
rotational	m±std	-0.02±1.56	-0.01±1.69	-0.02±2.02
apicobasal	med±iqr	-0.02±1.45	-0.05±1.58	0.11±2.35
rotational	med±iqr	-0.02±0.80	-0.02±0.88	-0.02±1.29
transventricular	erate	0.02	0.03	0.02
transmural	erate	0.22	0.23	0.25

computer tomography scan and taken as the ground truth excitation origin in the following. This study was approved by the institutional ethical board of the respective hospital (see [124]). Written informed consent was obtained from each patient.

The spatially Laplace interpolated pacemaker patient BSPMs of size 224x350 were presented to ScaleNet. Start and end of the ventricular depolarization were detected, signals were cut and re-sampled as with the simulated data. These BSPMs were again interpreted as images (two examples are shown in Figure 10.12) as before and were used as inputs for all LocaNets. The results are visualized in Figure 10.13 and summarized in Table 10.5. Compared to the errors from Table 10.3, localization errors are increased by more than 30 times. In contrast to the simulated results, RegressionNet performed best whereas FuzzyNet303 performed worst. Results for the single patients are shown in Table A.3. The single methods perform differently well on the single patient signals (Table A.3). This might speak for different features obtained from the input BSPMs. Two exemplary results of FuzzyNet303 and the respective ground truth are visualized in Figure 10.15: 007_LV performed bad with a localization error of 102 mm, 002_RV performed better with an error of 6 mm. The maximum value (coloring of the triangles) in 007_LV is smaller than in 002_RV. This could reflect an “uncertainty” of the network.

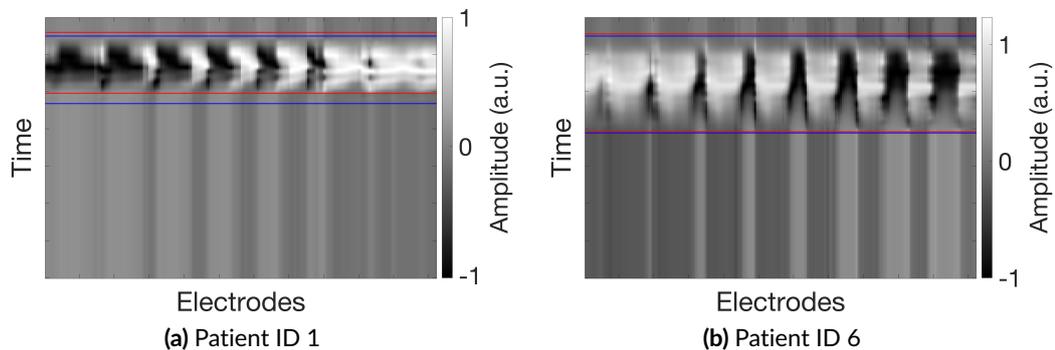


Figure 10.12: ScaleNet applied to patient data. Estimated start and end are shown as red horizontal lines. The signals were padded at the beginning and the end. The blue lines determine the bounds for an introduced padding as the signals were already cropped by clinicians.

Table 10.5: Localization errors (geodesic distance between prediction and label) as mean (m) and standard deviation (std), and as median (med) and interquartile range (iqr). All values are in mm.

Measure	FuzzyNet303	FuzzyNet93	RegressionNet
m±std	52.55±31.25	47.67±29.01	38.39±21.00
med±iqr	47.04±53.71	40.98±41.17	37.00±31.04

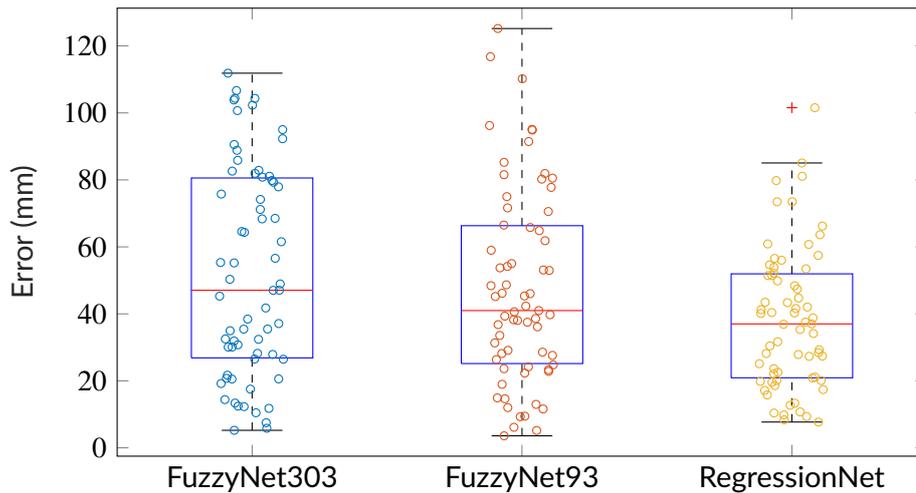


Figure 10.13: Boxplot of the errors (geodesic distance between prediction and label) of the LocaNets. The circles visualize the single errors.

FuzzyNet303 and FuzzyNet93 output a class prediction for each class. Thus, it is possible to calculate the top-three errors as the two networks output multiple possible solutions. The statistics of the top-three localization errors for the two FuzzyNets are shown in Table 10.6. The second/third most probable solution comes from the triangle with the largest sum of vertex values that is not connected to the first/second selected solution. Results generally improve, in the case of FuzzyNet93, the median drops down to 26 mm. The resulting boxplots are given in Figure 10.14.

Table 10.6: Top-three localization errors (geodesic distance between prediction and label) as mean (m) and standard deviation (std), and as median (med) and interquartile range (iqr). All values are in mm.

Measure	FuzzyNet303	FuzzyNet93
m±std	37.45±25.42	31.66±22.35
med±iqr	30.43±38.81	26.04±22.96

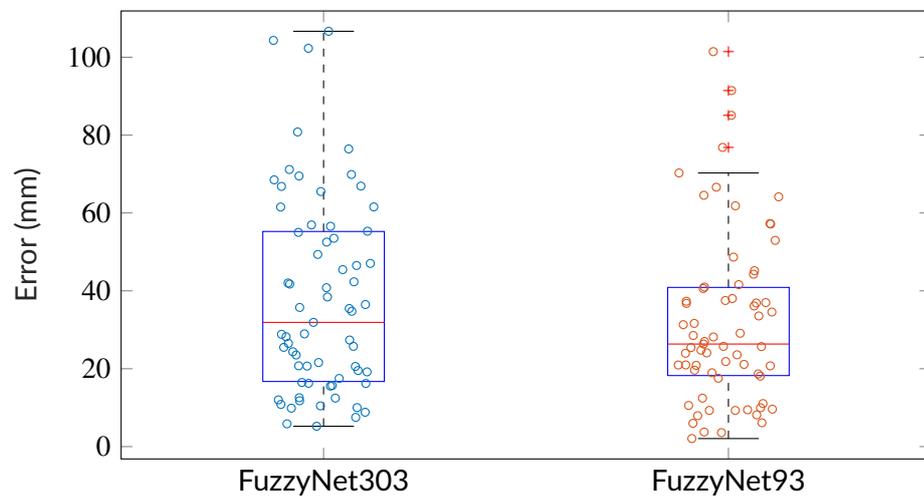


Figure 10.14: Boxplot of the top-three errors (geodesic distance between prediction and label) of the FuzzyNets. The circles visualize the single errors.

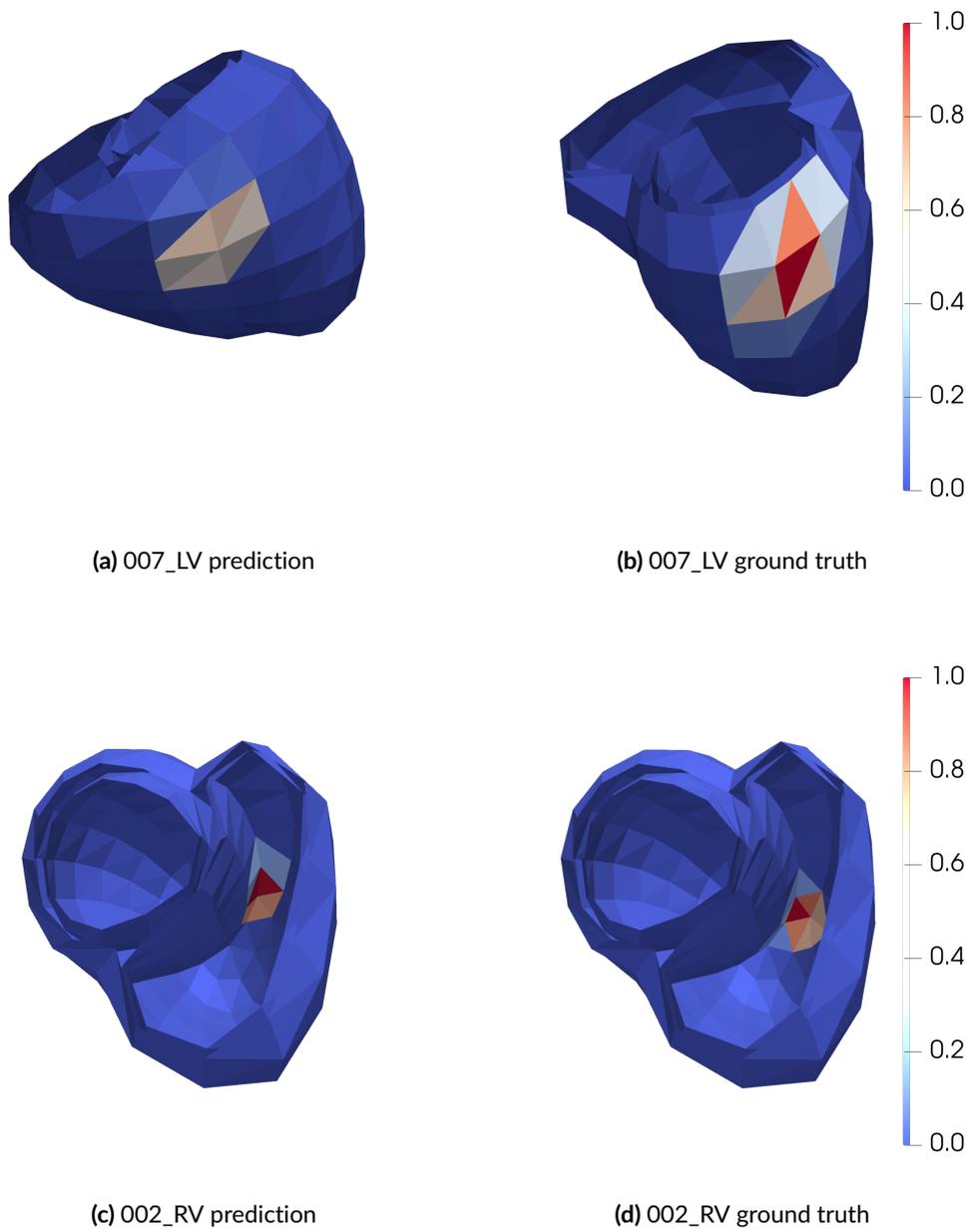


Figure 10.15: Two exemplary results from the patient dataset obtained with FuzzyNet303. Besides the prediction, the ground truth is shown. Results are visualized on a general classes geometry. Triangles are colored according to the sum of vertex values. The triangle with the largest sum was used for the calculation of the coordinates of the excitation origin.

10.6 Discussion

10.6.1 Data Generation

For this study, approximately 1.8 million BSPMs were generated. For being able to perform such a big number of simulations, less detailed modeling was requisite. Eikonal simulations paired with the boundary element method for forward calculation decreased the computational costs enormously compared to monodomain simulations with finite element method-based forward calculations. In contrast to the methodology used in Section 5.4, an optimization of the trigger points respectively the g_{K_s} gradient was not necessary since ectopic foci were simulated with only one triggering source and T waves were not evaluated. Although the QRS complex was expected to include most of the information of the location of triggering source, it cannot be excluded that the T wave does not contain any valuable information. Nevertheless, including the simulation of a realistic T wave would have consequenced in increased computational costs. The same holds for the introduction of an inhomogeneous torso. As already stated in Section 5.4, this would increase computational costs and also cause the problem of adequately choosing the conductivities of the different organs. Furthermore, it is not clear if the introduction of inhomogenities really leads to more realistic BSPMs [126]. The main advantage of the selected torso model, i.e. torso with ventricles and blood, was the easy introduction of different heart geometries into the same torso. This enabled a rotation and translation of the heart geometries without the need to adjust the surrounding compartments. Further variations have been introduced which will be discussed in the following: Conduction velocity of the ventricular tissue was varied between every BSPM. Therefore, the underlying activation time sequence from the Eikonal simulation was temporally scaled. This delivers the same results as changing the conduction velocity in the Eikonal simulations, namely changing the width of the QRS complex. Moreover, in this study, two unpublished algorithms were used: an adjusted method for the determination of ventricular fibers and a new formulation of ventricular coordinates. In empirical studies, they showed more consistent results. Nevertheless, the advantage of those still needs to be proven in systematic comparative studies. Next, the ventricular shape model is expected to cover most of the variations that can be observed in humans. Of course, it cannot be guaranteed that these cover all variation in a clinical setting which are needed for the training of a generalizing machine learning method. For example, the created heart geometries always had a right ventricular wall thickness of 2 mm which does not reflect the variation in reality. It might be that certain variations that would be important for the training of the learning algorithm were not taken into account. This is also valid for torso variations. In this study, one torso was used to generate all BSPMs. Although this might seem to be a problem, different other studies confirmed that the influence of the exact torso geometry is negligible compared to the variations of the heart geometry [127].

Last, the preparation of the data for the different learning methods will be discussed. Data was subdivided into training, validation and test dataset using 1000 geometrical variations

of the ventricles. This should guarantee the independence of training and test data. The extracted 200 lead BSPMs were interpolated to a 224 lead BSPMs obtaining a symmetrical grid of electrodes. This does not deliver any new information, however, facilitates the generation of symmetrical BSPM images. Apart from this interpolation, the introduction of the classes describing the excitation origin was a step during data preparation. The ventricular surfaces were subdivided into 560 triangles whose vertices formed 303 classes. The fuzzy class weights describing the position of the focus were obtained with the help of barycentric coordinates. This allows the description of each position on the ventricular surface. The number of classes was arbitrarily set to 303 which is not guaranteed to be optimal. Other numbers could be evaluated to decrease the number of parameters in the neural network and increase the training speed.

10.6.2 ScaleNet

The proposed method proved to be able to estimate the start and the end of a depolarization with a mean absolute error (on the test data) of 0.29 ms and 0.44 ms respectively. However, different aspects need further discussion: First, although the network selection was well-thought-out, it was not compared against other architectures. ResNets (and therefore ResNeXt), were designed to be able to overcome the problem of vanishing gradients in deep neural networks, which leads to a failure in learning, by introducing skipping layers [128]. Nevertheless, there are further architectures that might be appropriate and could deliver better results. The same holds for the choice of the hyperparameters which were chosen with special carefulness. Especially the learning rate is known to have a huge impact on the result. Although it was tried to use a reasonable learning rate strategy, it is not clear if this is the optimal choice. However, the optimization of the hyperparameters for CNNs, which is still a relevant research topic, often follows a trial-and-error procedure. Although, frameworks for the automatic optimization of hyperparameters exist [129], this is still computationally costly and therefore very time consuming. Consequently, the hyperparameter optimization needed to be constrained due to time constraints.

Moreover, the influence of the shape of the input data was not evaluated. It might be that a three dimensional input shape (i.e. not going around the torso spirally but preserving the arrangement of the electrodes in two dimensions and adding time in a third dimension) might deliver better results. Adding one dimension goes however hand in hand with an increase of complexity as the filter kernels are three dimensional (when preserving the size of the original two dimensional kernel). Nevertheless, this could deliver a better result [89]. Furthermore, long short-term memory (LSTM) networks [130] which treat the temporal dimension in a special way, could help to improve results.

Third, it was not possible to systematically evaluate the performance of the network on patient data as the ground truth for the real start and end of the depolarization can hardly be determined and these data were not available.

Apart from these points, ScaleNet was only designed to deliver results that are useful for the following LocaNet. Instead of using the whole BSPM beat for the localization of the ectopic

foci in LocaNet, ScaleNet was introduced to be able to add a-priori knowledge to LocaNet. The information about the excitation origin is likely located in the QRS complex and the simulated repolarization wave was not realistic (as mentioned above), so ScaleNet helped to reduce the information to the depolarization. Hence, the complexity of the task for the following LocaNets should be reduced, too.

Finally and in spite of all these points, ScaleNet could be rated as a success since the LocaNets were successfully applied to the problem.

10.6.3 LocaNets

All LocaNets showed median localization errors from 1.5 mm to 2.5 mm on simulated test data. Against the background of the huge geometrical variations, the achieved errors are noteworthy and this approach can be rated as a precise localization method. The results could further lead to the conclusion that the information of the patient geometry is not needed. Alternatively, the geometry information might be only necessary for the application of the method to pathological BSPMs as the results on clinical data might indicate. During this project, it was not possible to finally answer this question.

As already in the case of ScaleNet, the choice of the network architecture, the exact learning rate strategy and the influence of the input shape were not compared against other approaches. FuzzyNet303 performed best in the case of simulated signals. This might be related to the loss function. During the optimization, FuzzyNet303 uses only one kind of categorical cross entropy. With RegressionNet and FuzzyNet93, however, a combination of mean squared error and binary cross entropy, respectively a combination of binary and categorical cross entropy was used. It could be necessary to optimize the weighting of these different loss terms (mean squared error, cross entropy) which was not done in this project.

The application of the FuzzyNets provided the possibility of visualizing different solutions as with classical ECG imaging (ECGI)-based methods. Therefore, the ventricular surface was subdivided into 560 triangles whose 303 vertices were used as classes. The class weights came from the barycentric coordinates describing the focus in the respective triangle. This enabled the combination of regression and classification. Nevertheless, it was not proved if the choice of 303 triangles was optimal for this learning task. Results could improve if the number of classes (and thus the dimension of the optimization problem) was changed.

10.6.4 Application to Clinical Data

ScaleNet and LocaNets were applied to patient data without any adaption. Hence, the networks were never trained with a patient signal. Keeping this in mind, the median localization error of 37 mm to 47 mm is already noteworthy. Considering the top-three estimated excitation origins with the FuzzyNets, the median errors went down to 26 mm and 30 mm, respectively. These errors were achieved without a patient-specific geometry, any patient-specific adjustments or similar steps that are common when applying ECGI

for the localization of the excitation origin. Including all this information, the ECGI-based estimation achieves median localization errors in the range of 5 mm to more than 30 mm [124], but evaluating only 10 patients. Unfortunately, it is not known which patients were selected. Even works utilizing CNNs use the patient geometries to evaluate their results on clinical data [112, 113]. To the best knowledge of the author, this was the first time, a CNN for the localization of the excitation origin was trained only on simulated data and then directly applied to patient data.

As shown in Figure 10.13, in some clinical examples, errors could be up to 120 mm. It is not finally clear, why the networks failed to predict the excitation origin in these cases. Problems could be related to the prediction of ScaleNet which could not be evaluated on clinical data, to a design error in the LocaNets, or to a problem with the patient data. A design error of the LocaNets is however unlikely since they performed well on simulated data, whereas there was one big issue that was present in the dataset: The BSPM data were already cropped by clinicians. They usually determine QRS onset and offset and shorten the signal using these points. This could lead to missing information. Actually, in some cases, ScaleNet detected the start of the depolarization before the signal started. This might indicate wrong cropping. Moreover, the data pre-processing could alter the results. As with the simulated data, the BSPMs of the patient dataset were normalized lead-wise. Thus, noise is amplified. Although the results on simulated data which were artificially disturbed with measured noise, suggest that this should not be a problem. Nevertheless, this influence on the results cannot be totally excluded.

Furthermore, it should be emphasized that the pacemaker electrodes were naturally not equidistantly distributed over the ventricles but often located in the same region which might add a bias to the results. Additionally, the pacemaker data were obtained from sick patients, otherwise they would not have an implanted pacemaker. The possible underlying structural heart tissue changes (e.g. areas of infarction scar) were not included in the simulations. These, however, can lead to altered signal morphologies. One possibility to compensate this is a further cropping of the signals as e.g. done in [112, 113]. Nevertheless, this means the reduction of information presented to the network. A better trade-off between these facts could lead to better results.

Last, the training of the LocaNets was stopped after 48 hours. Although results on simulated data were in the range of 1 mm to 3 mm, this might be too short to use the networks on clinical data. It might be that the LocaNets continue learning to better extract the needed information from noisy input data.

10.7 Outlook

There are a couple of open points arising from these results. The most promising will be listed here to give a roadmap for further steps. They are mainly related to the identification of the reasons for the errors during the application to clinical data. One possible problem can be tackled by continuing the training of the single LocaNets. This could lead to further

improvements in the results. Moreover, another input shape of the BSPMs, e.g. a three-dimensionally arranged input array, might deliver better results as already shown in [89]. Furthermore, the design of the classes and therefore the learning tasks could be changed: 303 classes on the epicardial and endocardial surface might be too many outputs for a successful training during the given time. This is substantiated by the fact that RegressionNet with five outputs showed the most robust results on the clinical dataset. A successive merging of the 303/91 triangles could lead to better results. An alternative explanation might be that the models are too overfitted to simulated data. Lower SNR values in the dataset, missing electrodes, wrongly cropped BSPMs, and further real-world problems could be introduced into the simulated dataset and therefore into the training.

Last, the training of the networks should be resumed with patient data. It might be that just a few epochs improve the results dramatically as the networks learn to analyze patient data, too. A publication of the most important algorithms as in Part IV is planned.

Conclusion

In this part, a proof of concept study based on one patient geometry motivated the application of supervised learning strategies to the problem of the localization of the excitation origin in the ventricles. In a next step, convolutional neural networks (CNNs) were applied to localize the source in the ventricles based on an extended dataset of simulated body surface potential maps (BSPMs) extracted from 1.8 million different heart geometries with different orientations in the torso. The median localization errors on simulated but noisy data were below 2.5 mm. This approach could be applied to patient data without additional training on patient data. Although median estimation errors were between 37 mm and 47 mm (top-three median error 26 mm to 37 mm), the results can be compared with those from ECG imaging (ECGI)-based methods. Only trained with simulated data, without the use of patient-specific information, and without any further optimization of the methods to better work on the clinical signals, these results underline the potential of the method. With additional optimized network structures, learning (optimization) strategies, and clinical data inputs, the results are likely to further improve. This yields a generally applicable method being able to accelerate the ectopic foci localization procedure during a catheter lab intervention and thereby decrease the intervention time.

PART IV

OPEN SOURCE PROJECTS

Introduction to the Topic

It is very common to publish research results in scientific journals or present them at conferences. This opens the possibility to present the findings to a greater audience. Nevertheless, especially in the biomedical sector, many studies depend on patient or experimental data and therefore, the results cannot always be reproduced [131]. Furthermore, it is not uncommon that published studies lack a complete description of the methodology which hinders a repetition of the study. These two points are usually embraced by the terms “reproduction crisis” or “replication crisis” [132]. Although it is nearly impossible to guarantee the reproducibility during data acquisition in a clinical environment, an adequate description of the computational methods is possible. Nevertheless, the danger of unintentionally leaving out information important for the implementation remains [133]. Publishing the (source) code helps to overcome this issue. Alongside, the code can be reviewed by the research community and even be evaluated on further data. All these points are encouraging arguments to publish the source code for the main parts of this thesis in public repositories, e.g. on github. In this part, two open source projects born out of this thesis will be presented. The methodology behind this projects was mainly described before in this or in other articles [134–138]. However, the performance of the published algorithms will be evaluated. Moreover, the impact and uniqueness of the projects will be described in this chapter.

ECGdeli - ECG Delineation Algorithms

13.1 Introduction

In recent years, the amount of collected medical data has been increasing significantly. This phenomenon holds for the electrocardiogram (ECG) as well. The analysis of these signals is commonly done automatically to avoid time consuming manual work for physicians and to increase the capacity in the clinics. Moreover, a retrospective review allows for optimizing diagnosis, treatment, drug development, and the development/validation of new ECG processing methods as in this work [139–144]. Usually, these automatic approaches rely on the detection of the wave types (P waves, QRS complexes, and T waves) and their boundaries in the ECG which is often referred to as ECG delineation. The calculation of simple but clinically important temporal parameters like QT intervals, RR intervals, etc. as well as advanced features like slopes, heart rate variability, etc. use the detected points as input. This was also the case for the methods used in the studies presented in part II.

The algorithms for ECG processing and wave detection were developed and improved during this and a preceding [145] project. The step towards reproducible scientific results suggests publishing these algorithms as the results presented in this thesis depend on these algorithms. Hence, a toolbox for MATLAB was published under the open source license GPLv3 on github containing the ECG delineation algorithms. This package is called ECGdeli.

ECGdeli is not the only freely available ECG processing toolbox. Two further implementations available on PhysioNet are *ecg-kit* [146] and *ecgpuwave* [147]. In contrast to these software packages, ECGdeli offers a highly modular structure and focuses on a concrete purpose: ECG delineation. File input/output, feature extraction and further post processing is intentionally skipped or outsourced into other projects to keep the project as lean as possible. In the following, ECGdeli and its performance will be presented and compared to existing implementations. The impact of this software package will be explained.

This project was published in [148]. The repository was furthermore published on the open source platform zenodo [149] and is accessible on github.

13.2 Performance

ECGdeli contains filtering algorithms and wave delineation algorithms for the QRS complex, the P wave, and the T wave. Furthermore, a voting algorithm can be used to align the results for single leads and get a multi-lead delineation result. The performance was rated using the QT database [120, 150]. The QT database consists of 105 ECG recordings with two leads each. Beat annotations by clinicians for P waves, QRS complexes and T waves were extracted from the files with the suffix `*.q1c` and `*.q2c`. Detection errors, i.e. the temporal difference between detection and annotation, were calculated for both leads separately. Afterwards, for each annotation, the best result (lowest error) was chosen. As the clinicians had always both leads by hand during annotation, it was not possible to determine on which lead the physician's decision was based on. Therefore, this seemed to be the most objective way to evaluate the results.

The performance of ECGdeli was compared to `ecg-kit` [146] (version 1.0, Commit `c8e3de4` on github) and `ecgpuwave` [147] (version downloaded on 01/08/2020 from `physionet`). `Ecg-kit` offers different methods for wave delineation. The option `wavedet` was selected to obtain the peaks and wave boundaries for P waves, QRS complexes, and T waves. It should be emphasized that the exact same way of calculating the errors was applied for all methods. For comparison, mean (absolute) errors, their standard deviation and median errors and the interquartile ranges of the errors were determined. The scripts for error calculation were published in a CodeOcean project (doi 10.24433/CO.9115981.v1) to allow the reproduction of the results and evaluation code review.

Table 13.1 shows the results of this evaluation. Errors of all approaches are in a comparable range. Dependent on the statistical measure, one method outperforms another. No consistent trend was visible. For T waves, ECGdeli performs better regarding the interquartile range and most of the other measures, but for QRS complexes, the other methods outperform ECGdeli. A first assumption that ECGdeli might deliver better results when less annotations are considered could not be confirmed. When considering only the points found by all three methods, results do not change a lot (Table A.2). Nevertheless, the errors of ECGdeli might also be related to the missing detection of biphasic and M-shaped P and T waves. ECGdeli always detects a monophasic P and T waves. This classification could be performed in a post processing step and further improve results.

13.3 Impact

Publishing the ECG delineation algorithms in the open source package ECGdeli meets several needs. First, several published studies used the algorithms in the current or slightly different form [134–138]. Furthermore, ECGdeli was used in this work to find the wave boundaries important for template generation and single lead processing. Releasing the algorithms, is an important step towards reproducibility of the studies presented here.

Second, ECGdeli can now be used as a method for comparing new or closed source algorithms. This can always be done even without an open source package by calculating the errors on standard databases like the QT database and comparing them to given values in the respective publications, e.g. from [151, 152]. However, slight differences in error calculation can influence the results making them in practice hardly comparable. Moreover, standard databases might be biased not representing pathologies that could be relevant for a desired application and should be considered for an evaluation, too. This is why releasing the benchmark algorithms is surely a crucial step towards better comparability. Alternatively, at least the methods for calculating the errors can be released. Both was done with ECGdeli. Finally, the simple and modular structure of ECGdeli promotes a possible incorporation of the algorithms into existing projects. This is admitted by releasing the code under the GNU Public Licence (GPLv3) allowing adaption, modification and redistribution as long as the code is shared again.

Table 13.1: Detection errors of ECGdeli, ecg-kit and ecgpuwave compared to manual expert annotations in samples as well as number of detected points/available annotations. The number of detected points refers to the number of waves that were also annotated, so it can be maximum the number of annotations. med: median, iqr: half interquartile range, m: mean, std: standard deviation, signed: signed errors, abs: absolute errors.

	P Onset	P Peak	P Offset
med±iqr ECGdeli	2.00±3.00	1.00±2.00	3.00±4.00
m±std abs ECGdeli	3.87±6.14	2.46±5.70	4.17±6.38
m±std signed ECGdeli	-1.51±7.10	0.65±6.18	2.26±7.27
med±iqr ecg-kit	2.00±4.00	1.00±2.00	2.00±3.00
m±std abs ecg-kit	3.33±6.66	2.71±6.57	3.30±5.64
m±std signed ecg-kit	-1.10±7.36	-1.17±7.01	-2.18±6.15
med±iqr ecgpuwave	3.00±5.00	2.00±2.00	2.00±4.00
m±std abs ecgpuwave	4.52±6.05	2.96±4.62	3.89±4.91
m±std signed ecgpuwave	3.33±6.78	-0.24±5.48	-1.23±6.14
detected points ECGdeli	3194	3194	3194
detected points ecg-kit	3096	3096	3096
detected points ecgpuwave	2127	2127	2127
number of clinical annotations	3194	3194	3194
	QRS Onset	QRS Peak	QRS Offset
med±iqr ECGdeli	2.00±4.00	1.00±1.00	3.00±4.00
m±std abs ECGdeli	4.01±8.50	2.46±8.46	4.39±8.53
m±std signed ECGdeli	-1.31±9.30	-0.60±8.79	2.19±9.34
med±iqr ecg-kit	1.00±4.00	2.00±2.00	1.00±3.00
m±std abs ecg-kit	1.66±2.14	3.36±4.37	1.91±2.40
m±std signed ecg-kit	0.54±2.65	-3.13±4.54	-0.46±3.04
med±iqr ecgpuwave	2.00±4.00	3.00±2.00	2.00±3.00
m±std abs ecgpuwave	3.46±3.55	3.54±4.32	3.51±5.48
m±std signed ecgpuwave	-1.61±4.69	-1.71±5.31	-0.38±6.49
detected points ECGdeli	4019	4019	4019
detected points ecg-kit	4019	4013	4018
detected points ecgpuwave	4018	4019	4017
number of clinical annotations	4019	4019	4019
	T Onset	T Peak	T Offset
med±iqr ECGdeli	5.00±9.00	2.00±3.00	3.00±7.00
m±std abs ECGdeli	8.86±9.98	6.24±12.29	7.18±11.24
m±std signed ECGdeli	-3.65±12.84	-0.72±13.76	-1.92±13.20
med±iqr ecg-kit	5.00±14.00	1.00±14.00	3.00±11.00
m±std abs ecg-kit	9.82±13.43	4.62±12.24	6.67±14.38
m±std signed ecg-kit	-1.11±16.60	-2.18±12.90	-2.81±15.60
med±iqr ecgpuwave	8.00±14.00	2.00±14.00	6.00±11.00
m±std abs ecgpuwave	13.10±13.98	10.50±17.66	11.57±16.56
m±std signed ecgpuwave	9.96±16.37	7.51±19.12	3.86±19.84
detected points ECGdeli	1414	3936	3936
detected points ecg-kit	1332	3801	3812
detected points ecgpuwave	1315	3865	3864
number of clinical annotations	1414	3936	3936

ECGfeat - ECG Feature Extraction Algorithms

14.1 Introduction

As already stated in Chapter 13, the automatic analysis of the electrocardiogram (ECG) is crucial for tackling the problem of analyzing the increasing amount of available health data. Next to ECG delinention, ECG feature extraction is the base of many ECG processing studies. These can comprise the classification of ventricular extra beats [73], the estimation of the potassium concentration [56, 57], the assessment of mental workload [153], and many more topics. However, underlying feature extraction algorithms are commonly neither published nor evaluated regarding their robustness. To allow the adaption of the algorithms to similar problems, both a transparent description and an evaluation of the robustness are crucial. This is why the feature extraction algorithms used in this work were evaluated similarly to Section 5.2.1 regarding their robustness against noise and published online on github under the GPLv3 license. This allows a reproduction of the results presented in this work as well as benchmarking new algorithms with the proposed published evaluation strategy and data. Furthermore, the proposed methods can be adapted to new problems. In addition to all mentioned advantages, a possibly missing description of important details in the implementation is intrinsically excluded.

In the following, the contents of the github repository will be explained. Moreover, the impact of this repository is reviewed. The methodological description of the feature calculation will be skipped as this was already done in Section 5.2.1.

Besides the algorithms presented here, further feature extraction techniques were added to the repository described and presented in [145]. They were not benchmarked regarding their robustness, neither in this nor in a preceding work.

14.2 Contents

The repository contains a function for artifact detection `artifact_detection.m`, a function for ECG beat template generation `template_generation.m` and a feature extraction algorithm `feature_extraction.m` calculating the features presented in Section 5.2.1. Furthermore, two examples are included: first, the source code from the evaluation presented in the next section. Second, an example with a clinical signal from the PTB database [93, 120] showing one possible workflow using ECGdeli and ECGfeat.

The following features can be calculated:

- F1 first statistical moment of the T wave distribution
- F2 second statistical moment of the T wave distribution
- F3 third statistical moment of the T wave distribution
- F4 fourth statistical moment of the T wave distribution
- F5 RT distance (R peak to T peak)
- F6 RT mid distance - in the case of biphasic waves
- F7 peakedness of the T wave
- F8 T wave amplitude
- F9 slope of the ascending part of the T wave
- F10 slope of the descending part of the T wave
- F11 ratio of first half T wave energy and whole T wave energy
- F12 ratio of second half T wave energy and whole T wave energy
- F13 R peak amplitude
- F14 R peak energy
- F15 ratio R peak energy and R peak amplitude
- F16 ST segment change (elevation or depression)
- F17 flag for biphasic T waves (0: monophasic, 1: biphasic)
- F18 R peak area under curve

In the following, the abbreviations F1-F18 will be used. The repository is available under <https://github.com/KIT-IBT/ECGfeat>.

14.3 Pre-processing Recommendation and Robustness Evaluation

The simulated noise-free 12-lead ECG signals introduced in Section 5.2.1 were filtered with a phase-free Butterworth filter of fourth order (available in ECGdeli) with different cut-off frequencies for high-pass (0.05 Hz to 0.5 Hz) and low-pass (20 Hz to 250 Hz). By doing so, recommendations for the pre-processing could be determined as described in the following: All features were calculated on the filtered noise-free signals and the unfiltered noise-free

signals without any averaging of the single beats. If the filtering does not influence feature extraction, these two feature value sets must be equal, so the relative deviation of both should be minimal. Hence, the maximum mean absolute deviation of the two feature sets among all leads was calculated to find the “worst case lead”. Deviations were normalized to the feature value from the unfiltered dataset. This is formulated in the following equation

$$\text{dev}_i = \max_l \frac{1}{N} \sum_{n=1}^N \frac{|(f_{i,l,\text{filt}}(n) - f_{i,l,\text{unfilt}}(n))|}{f_{i,l,\text{unfilt}}(n)}, \quad (14.1)$$

where dev_i is the deviation for feature i , l is a specific lead, N is the number of signals, $f_{i,l,\text{unfilt}}/f_{i,l,\text{filt}}$ is the feature value of i in l from the unfiltered/filtered ECG signal. The lowest (low-pass) or highest (high-pass) frequency with a maximum dev_i of 5% $\text{dev}_{i,5\%}$ was selected as the respective filter bound. As F17 is the result of a binary classification, dev_i was replaced by the minimum of the accuracy which determines a change of the classification result.

As expected, cut-off frequencies for features calculated from the QRS complex (F13-15, F18) showed higher cut-off frequencies for both filters (Table 14.1) as the spectrum of the QRS complex is located at higher frequencies [154].

Table 14.1: Recommendations for the cutoff frequencies (cf) of a low-pass (LP) and high-pass (HP) filter. Furthermore, $\text{dev}_{i,5\%}$ is given for each feature and filter type in percent. For F17, the accuracy (Acc) is given in percent.

	F1	F2	F3	F4	F5	F6	F7	F8	F9
LP cf	20	40	40	40	20	20	40	40	40
LP $\text{dev}_{i,5\%}$	3.81	0.37	0.45	0.52	1.68	1.87	0.75	2.18	0.76
HP cf	0.05	0.10	0.10	0.10	0.05	0.05	0.10	0.10	0.10
HP $\text{dev}_{i,5\%}$	3.81	0.37	0.45	0.52	1.68	1.87	0.75	2.18	0.76
	F10	F11	F12	F13	F14	F15	F16	F17	F18
LP cf	40	40	60	70	50	60	40	40	50
LP $\text{dev}_{i,5\%}/\text{Acc}$	0.74	0.61	3.22	4.12	3.76	4.37	1.41	1.00	3.04
HP cf	0.10	0.10	0.30	0.40	0.20	0.30	0.10	0.05	0.20
HP $\text{dev}_{i,5\%}/\text{Acc}$	0.74	0.61	3.22	4.12	3.76	4.37	1.41	1.00	3.04

For the assessment of the robustness, noisy ECG signals were generated by adding white Gaussian noise to the noise-free simulated ECGs used before obtaining a signal to noise ratio (SNR) of 10 dB, 20 dB, or 30 dB. This was repeated 50 times per SNR introducing different realizations of the noise. Features were extracted from the noisy signals after filtering with the filter boundaries from Table 14.1. Normalized deviations of feature values of the noisy filtered $f_{i,l,\text{noisy}}$ from the unfiltered signal were calculated using

$$\text{dev}_{i,l}(n) = \frac{|(f_{i,l,\text{noisy}}(n) - f_{i,l,\text{unfilt}}(n))|}{f_{i,l,\text{unfilt}}(n)}. \quad (14.2)$$

Medians and interquartile ranges of all $n = 1, \dots, N$ deviations were evaluated considering all leads l . As Table 14.2 shows, the error change from 10 dB to 20 dB is larger than from 20 dB

to 30 dB for almost every feature. This suggests that the user should aim at achieving an SNR of at least 20 dB for better results. Although the slope features (F9-F10) were designed to deliver higher robustness through the polynomial fit (Section 5.2.1), the deviations (regarding the interquartile range which considers 50% of the deviation values around the median) could be still as high as 22.9% to 38.3% for 30 dB. It should be emphasized that this error can be reduced dramatically by averaging over the feature values from the 50 noisy ECGs. For F9, for example, an error of $1.1\% \pm 1.2\%$ could be achieved. As expected, the robustness increases for a repeated measurement and should therefore be implemented. Besides feature averaging, other techniques to increase the SNR, e.g. beat averaging or lead reduction, can increase the robustness as well (Section 5.2.1).

Table 14.2: Relative errors in percent between the features from the noisy filtered signals and the features from the noise-free unfiltered signals over all realizations. The binary F17 was evaluated by calculating the accuracy in percent. Values are given as median \pm half interquartile range.

	F1	F2	F3	F4	F5	F6
10 dB	4.7 \pm 3.5	7.7 \pm 5.4	11.1 \pm 7.7	15.4 \pm 11.3	11.9 \pm 7.2	11.8 \pm 7.1
20 dB	4.0 \pm 3.1	6.1 \pm 4.2	8.2 \pm 5.5	10.0 \pm 6.7	11.8 \pm 7.1	11.7 \pm 7.1
30 dB	3.9 \pm 3.1	6.0 \pm 4.2	8.1 \pm 5.4	9.9 \pm 6.5	11.8 \pm 7.1	11.7 \pm 7.1
	F7	F8	F9	F10	F11	F12
10 dB	13.6 \pm 8.8	10.4 \pm 7.4	28.8 \pm 19.4	17.8 \pm 14.1	11.4 \pm 8.0	52.7 \pm 64.6
20 dB	10.2 \pm 6.5	8.4 \pm 6.3	23.6 \pm 16.5	12.6 \pm 11.3	9.3 \pm 6.3	34.7 \pm 44.5
30 dB	9.6 \pm 6.2	8.3 \pm 6.2	22.5 \pm 15.8	11.9 \pm 11.0	9.1 \pm 6.2	33.0 \pm 42.9
	F13	F14	F15	F16	F17	F18
10 dB	4.5 \pm 2.9	7.8 \pm 5.8	5.4 \pm 4.9	23.6 \pm 18.1	1.0 \pm 0.0	19.2 \pm 14.1
20 dB	4.0 \pm 2.6	7.0 \pm 5.4	4.8 \pm 4.7	18.4 \pm 15.9	1.0 \pm 0.0	16.0 \pm 12.5
30 dB	3.9 \pm 2.6	6.9 \pm 5.3	4.8 \pm 4.7	17.8 \pm 15.8	1.0 \pm 0.0	15.2 \pm 11.9

14.4 Impact

Known to the author, this is the first morphological ECG feature extraction toolbox evaluated regarding its robustness and providing recommendations for the application of bandpass filtering. Furthermore, it seems to be the first toolbox written in MATLAB for morphological ECG feature extraction available on github. The provided results inherently and transparently reveal the benefits and shortcomings of the approaches. This is particularly important if other algorithms depend on the results of the feature extraction. In these situations, an unexpected result can be traced back to errors during feature extraction. The evaluation method and the simulated ECG signals are provided in the repository enabling the repetition of the study. Furthermore and for the first time, a comparison of different methods for feature extraction is possible (as done in Section 5.2.1) using the published performance assessment algorithms. Moreover, this can be used as a test environment to assess changes in the methodology and reject them if they deliver worse results.

Publishing the presented feature extraction algorithms under the GPLv3 license on github enables the use, adaption and improvement of the algorithms by the community. Furthermore, a step towards reproducible scientific results from this and recent works [145] is taken.

ECGconc - ECG Concentration Estimation Algorithms

15.1 Introduction

In a final open source project, the proposed concentration estimation algorithms were uploaded to a github repository. Although the underlying electrocardiogram (ECG) data could not be uploaded to the repository, a final global estimation model fitted on all patients could. This embraces all necessary sources and variables to apply the model. Furthermore, all sources used to parameterize the model were added to the repository. As the algorithms were extensively described in Chapter 6, the description will be skipped here. The source code in the repository is published under the GPLv3 license.

15.2 Contents

All scripts necessary to follow the workflow from ECG recordings to estimated concentration values were published in the ECGconc repository. In a first step, ECG templates were built and features were calculated in the script `Prepare_Data.m`: ECGdeli was applied to the signals to find P waves, QRS complexes and T wave positions in the signal. Afterwards, the signal was split into four minute segments around predefined evaluation points. These segments were further analyzed with the function `Artifact_Detection.m` to find artifacts in the signal that were omitted during template generation (`Create_Templates_Class.m`). For each lead, an ECG template was generated. The lead templates were subsequently reduced to one lead using maximum amplitude transform (`Lead_Transform.m`). The resulting single lead template was analyzed with ECGfeat and results were saved. In a second step, the parameterized concentration model could be applied (`Apply_Conc_Model.m`) to the found features. The user should keep in mind that the model still needs concentration measurements from a blood test to apply the patient-specific correction.

Apart from this workflow for the application of the global model, the code for the generation of the patient-specific `Find_Conc_Model_Pat.m` and the global model `Find_Conc_Model_Global.m` was also added to the repository. The repository is available under <https://github.com/KIT-IBT/ECGconc>.

15.3 Impact

For the first time, an ion concentration estimation pipeline was published under an open source license. ECGconc contains all information to apply both presented approaches from Chapter 6. The step of publishing the results is mainly owing to a conclusion from [60]: All existing published approaches for concentration estimation (see Table 4.1 in Chapter 4) rely on non-public datasets. Thus, an exact comparison between the approaches is hardly possible. By releasing ECGconc under GPLv3 license, the burden of an application by data holders has now been lowered significantly. The existing approaches in ECGconc can simply be applied and no implementation work has to be done. If this happens, the methods from this work developed on the dataset coming from the University of Bologna could be applied to totally new data which could confirm or disprove the effectiveness of the proposed method. Depending on the dataset, a further effect might be that the algorithms could be applied to data by patient groups other than hemodialysis patients. This is extremely important for a further evaluation since it is not clear if the features useful in dialysis patients also fit to patients from other groups (Section 6.5). Moreover, the performance of new and existing approaches could be compared to ECGconc. This might also strengthen collaboration between the relatively small community doing research on concentration estimation from the ECG. Last, the release of this approach could encourage others to follow the example of ECGconc making the approaches publicly accessible.

As already ECGdeli and ECGfeat, ECGconc and the chosen license GPLv3 allows the adaption and improvement of the approaches in future work. On the long run, this might help to bring ECG-based concentration estimation into clinical practice.

Summary and Conclusion

Using source code from the studies presented in this thesis, three repositories could be published on github providing all details on the essential algorithms necessary for calculating the results. The first project is ECGdeli, containing all the electrocardiogram (ECG) filtering and delineation algorithms used in this work. As other open source delineation projects existed, they were compared to the presented one. It could be proven that ECGdeli at its current state delivers comparable results to the existing methods although there was no rejection criterion for leaving out single waves. This could further improve the results. The ECGdeli project was previously released and shows already 94 views of the repository and 7 clones in two weeks (11/23/2020 to 12/06/2020). This underlines the acceptance of the project.

The follow-up project ECGfeat includes morphological feature extraction algorithms compatible with the simple data structure introduced with ECGdeli. Algorithms are transparently assessed regarding their robustness against noise. Furthermore, recommendations for an adequate bandpass filtering not disturbing the feature values is given. These two points characterize this project as unique. Moreover, to the best knowledge of the author, this is the first toolbox for morphological feature extraction written in MATLAB available on github. Finally, the algorithms in the repository ECGconc include the algorithms of the proposed potassium concentration estimation method. This is the first time, the source code for the whole concentration estimation pipeline was published as open source project.

By publishing the algorithms on github under the open source license GPLv3, the test of the reproducibility of the results presented in this thesis is facilitated. Furthermore, the algorithms can be reviewed, adapted to new problems, and used as basis for follow-up studies or as benchmark for comparison.

PART V

FINAL REMARKS

Conclusion

In this thesis, two major studies were presented. In both, simulated electrocardiogram (ECG) signals helped to develop methods applicable to patient data. These optimized methods enabled the extraction of the following health information from the ECG: blood potassium concentration ($[K^+]_b$) and the position of an ectopic focus in the ventricles. The underlying algorithms were discussed and compared to existing approaches in detail in Parts II and III. In the following, a top-level view on the projects will be given with respect to the general aims of this thesis which were to support and accelerate making diagnoses to improve the patient outcome by using knowledge from data and simulations.

In the first presented project, a quick and reliable quantification of dyskalemia was desired for preventing cardiovascular events that are connected to pathological $[K^+]_b$ [60]. It was shown how simulations can help to detect possible problems that need to be considered during method development. These findings were subsequently incorporated into improved signal processing methods. Two workflows for $[K^+]_b$ concentration estimation, involving the optimized algorithms and for the first time published as open source projects, were developed achieving results as good as or better than proposals from literature. For the first time, the distribution of concentrations in the dataset was considered allowing an unbiased view on the results. With these powerful ECG-based algorithms for a quick non-invasive $[K^+]_b$ concentration estimation, the secondary prevention of cardiovascular events can be facilitated.

The automatic localization of ectopic foci with the ECG can help to decrease the intervention time by supporting the cardiologist in finding ectopic foci in the ventricles that need to be ablated. Again, ECG simulations were utilized. The presented deep learning approach was trained only with simulated data and successfully applied to patient data. This approach allows not only to localize the position of the ectopic focus but also to visualize localization uncertainties of the algorithm which can be highly beneficial during an intervention in the catheter lab.

In conclusion, ECG simulations paired with ECG signal processing can be utilized to deliver diagnosis support. Especially when using simple techniques for parameter estimation like linear regression, patient-specific variations cannot be compensated easily. Under the constraint in biomedical engineering of a large inter-patient variation and the lack of data, the

solution must be the application of a patient-specific approach. Alternatively, as shown in the second project, the generation of a simulated dataset containing millions of data representing a large variety of patients and the application of a complex estimation method can help to account for the large inter-patient variations which makes the method appropriate for the use with patient data.

Ideas for Future Projects

In this final chapter of this thesis, ideas for future projects will be collected reflecting the findings from this thesis. Moreover, further topics being part of my research will be discussed using results not presented in this thesis.

18.1 Machine Learning and ECG Signal Processing

Especially convolutional neural networks (CNNs) are becoming more and more popular in electrocardiogram (ECG) processing, e.g. for arrhythmia classification [155], dyskalemia classification [107] or the detection of myocardial infarction[156]. The obtained results often outperform classical approaches. Instead of directly exploiting this technique for a classification task, CNNs could be used for the correction of patient-specific influences on the signal by finding patient specific correction values straight from the ECG. As shown in Chapter 10, the CNN should be able to compensate for the influences of different heart geometries, orientations, conduction velocities, fiber orientations etc. since results were good on the test dataset with other realizations of all these parameters. If this compensation could be used directly, a standardized ECG could be constructed and subsequently used with traditional methods. This could be for example the global concentration estimation technique presented in Chapter 6. Alternatively, a feature correction or an unbiased feature extraction could be imaginable with CNNs. In contrast to the meta features extracted by a CNN during training before the fully connected layer, this should of course comprise features that can be understood by humans.

This last point is closely connected with the problem of explainable artificial intelligence. Especially in the medical sector, the problem of not knowing the influences on the result of a CNN is severe. Robustness of the algorithms is an important point since. This is why there have been studies within this work together with Maike Rees and Steffen Schuler regarding the explainability of the algorithms presented in Part III. The question about the importance of certain parts of the ECG signal for the CNN to solve a specific task arose. With the help

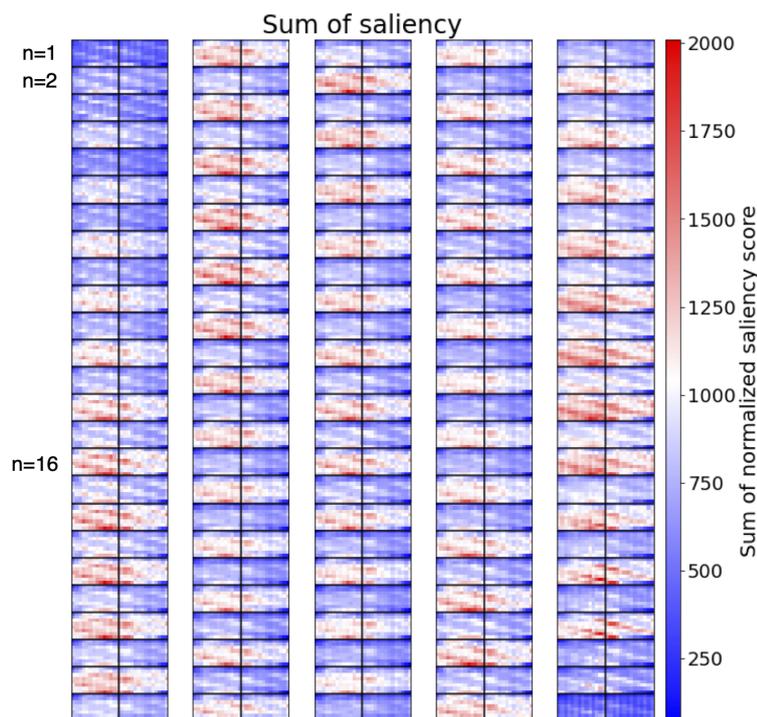


Figure 18.1: Importance map (saliency map [157]) of the body surface potential map. Temporal progression is visualized row-wise (n is the time step) and continued column-wise. The single pictures consist of two halves: torso front (left) and torso back (right). The right half of the picture was flipped to guarantee a continuous transition from front to back. Higher value represent higher importance.

of explainability algorithms, it was possible to start investigating on the importance of single electrodes for single tasks. This was exemplary done for a CNN classifying whether the activation started from the right or left ventricle (Figure 18.1). Electrodes on the torso front seem to be more important than on the back. Furthermore a diagonal “double-belt” (e.g. in timestep 16) is visible. This might be connected to the asymmetric location of the ventricles in the torso (the right ventricle partially covers the left in the anterior view). If the CNN wants to “see” also the left ventricle, it has to “look” from the electrodes labeled in red. Those are suspected to be the ones, for which the left ventricle is not covered by the right and are at the same time near the heart.

The explainability approaches could also be used to solve further problems regarding ECG signal processing, e.g. to determine the optimal number of leads and their placement on the torso for a specific task. This was already investigated with simulations for an optimal detection of myocardial ischemia [158]. However, a data-driven approach could be imaginable and beneficial for clinical acceptance.

In general, CNNs could be utilized to directly solve tasks that need several steps. One example apart from the localization of ectopic foci could be the estimation of activation times of the heart. A visualization of the direction of the excitation spread might deliver further information about the electrical activity of the heart. Most importantly but also most

challenging, an adequate underlying dataset showing the variations that need to be detected is the basis for a successful training of a CNN. Such a dataset could be generated with simulations following the results from Part III. Another possible application could be the determination of flutter mechanisms from P waves as done in the supervised work by Moritz Lindner [159]. Instead using recurrence plot analysis as in [127], the application of a CNN on the signals delivered first encouraging results. Nevertheless, the question remains open how flutter mechanisms can be grouped and if the classes introduced in [127] are optimal for this task.

18.2 Intracardiac Signal Processing

The measurement of signals from the atrial endocardium with a dedicated catheter is a common practice. During these diagnostic treatments, a multilead measurement of the excitation of the cardiac tissue can be performed. The interpretation of the signals which can usually be as many as 64 at a time is hard and the cardiologist needs experience and technical assistance. This is why intracardiac signal processing is one research topic at the Institute of Biomedical Engineering. During this PhD project, several projects related to intracardiac signal processing were started. As they were not described in this thesis, they will be mentioned here summarizing the ideas and possible follow-up projects.

The adequate pre-processing of intracardiac signals leaves space for improvement. Advanced techniques, like the proposed 3D Gaussian filter for the spatio-temporal filtering of optical mapping signals presented in [160] or the patented algorithm for ventricular far field compensation [161], could be applied. Furthermore, as shown in Chapter 14, the optimal filtering strongly depends on the used algorithms. A determination of filter parameters for the common algorithms applied in intracardiac signal processing seems to be relevant and could improve the obtained results.

The phase (angle) is a mathematical description of a periodic event. As certain atrial arrhythmias are characterized by periodic processes, the calculation of the phase is a standard technique to describe these recurrent processes. Different methods have been proposed, e.g. in [162, 163]. However, their robustness was never evaluated. In a study with Simon Süß, it was shown that most of the proposed methods do not deliver robust results when introducing small jitters in the periodicity, noise, and amplitude changes of the analyzed signals [164]. The calculation of a pseudo phase based on the maxima in the signal delivered the most robust results. The findings underline that the calculation of phase angles is inappropriate for the analysis of these periodic events since they usually show jitters, noise, etc. Similar studies regarding other standard processing techniques could be designed. This holds especially for local activation time detection which is the base of many follow-up steps in an intracardiac signal processing pipeline.

One example for the use of local activation times is the calculation of the conduction velocity on the atrial tissue. Typically, the proposed methods rely on the local activation times determined with a preceding algorithm. Regarding this, two projects together with Laura

Unger, Claudia Nagel and Jeanne Koch were completed. It was shown that most of the existing conduction velocity estimation methods inherently cannot deal with either wrongly annotated local activation times or local changes in conduction velocity [165]. This is why a-priori knowledge needs to be introduced and as expected, results in better velocity estimations [165, 166]. Analyzing the raw intracardiac electrogram signals, this knowledge can be presented and considered in the algorithms. Instead of only relying on precalculated local activation times [167], it was shown how to use the intracardiac signals to sort out inadequate local activation time estimates and consequently improve conduction velocity estimations. Nevertheless, further tissue properties like fibrosis, lines of block etc. need to be detected to further enhance conduction velocity estimation. Especially fibrosis is characterized by fractionated, i.e. chaotic, signal courses and a pathological conduction velocity. This complicates local activation time determination as there is no consistent definition of an activation time in these cases. This usually results in inconsistent activation times. The use of measures for describing the signal “complexity”, e.g. entropy measures, could help to introduce the needed a-priori knowledge to the conduction velocity estimation algorithm. These algorithms were utilized in the student research project of Rafi Beinhorn, in [168], and in [169]. The last step could be the incorporation of the conduction velocity estimation results into the virtual reality environment proposed in [170] for an appealing visualization.

Appendix

A.1 Himeno et al. Whole Heart Simulations

Table A.1: List of the performed simulations with the Himeno et al. model. The ID is an internal identifier for the respective simulation.

ID	[K ⁺] _o	[Ca ²⁺] _o	[Na ⁺] _o	ID	[K ⁺] _o	[Ca ²⁺] _o	[Na ⁺] _o
1	4.5	1.0	140	425	5.5	2.2	140
2	4.5	1.4	140	426	5.5	2.6	140
3	4.5	1.8	140	427	5.5	3.0	140
4	4.5	2.2	140	428	6.7	0.6	140
5	4.5	2.6	140	429	6.7	1.0	140
6	4.5	3.0	140	430	6.7	1.4	140
7	4.5	0.6	140	431	6.7	1.8	140
8	4.5	0.7	140	432	6.7	2.2	140
9	4.5	0.8	140	433	6.7	2.6	140
10	4.5	0.9	140	434	6.7	3.0	140
11	4.5	1.1	140	506	4.2	0.8	140
12	4.5	1.2	140	507	4.2	1.2	140
13	4.5	1.3	140	508	4.2	1.6	140
14	4.5	1.5	140	509	4.2	2.0	140
15	4.5	1.6	140	510	4.2	2.4	140
16	4.5	1.7	140	511	4.2	2.8	140
17	4.5	1.9	140	512	5.1	0.8	140
18	4.5	2.0	140	513	5.1	1.2	140
19	4.5	2.1	140	514	5.1	1.6	140
20	4.5	2.3	140	515	5.1	2.0	140
22	4.5	2.5	140	516	5.1	2.4	140
23	4.5	2.7	140	517	5.1	2.8	140
24	4.5	2.8	140	518	5.4	0.8	140
407	3.8	0.6	140	519	5.4	1.2	140
408	3.8	1.0	140	520	5.4	1.6	140
409	3.8	1.4	140	521	5.4	2.0	140
410	3.8	1.8	140	522	5.4	2.4	140
411	3.8	2.2	140	523	5.4	2.8	140
412	3.8	2.6	140	524	6.1	0.8	140
413	3.8	3.0	140	525	6.1	1.2	140
414	4.7	0.6	140	526	6.1	1.6	140
415	4.7	1.0	140	527	6.1	2.0	140
416	4.7	1.4	140	528	6.1	2.4	140
417	4.7	1.8	140	529	6.1	2.8	140
418	4.7	2.2	140	530	7.3	0.8	140
419	4.7	2.6	140	531	7.3	1.2	140
420	4.7	3.0	140	532	7.3	1.6	140
421	5.5	0.6	140	533	7.3	2.0	140
422	5.5	1.0	140	534	7.3	2.4	140
423	5.5	1.4	140	535	7.3	2.8	140
424	5.5	1.8	140				

A.2 Influence of Filtering on Feature Extraction

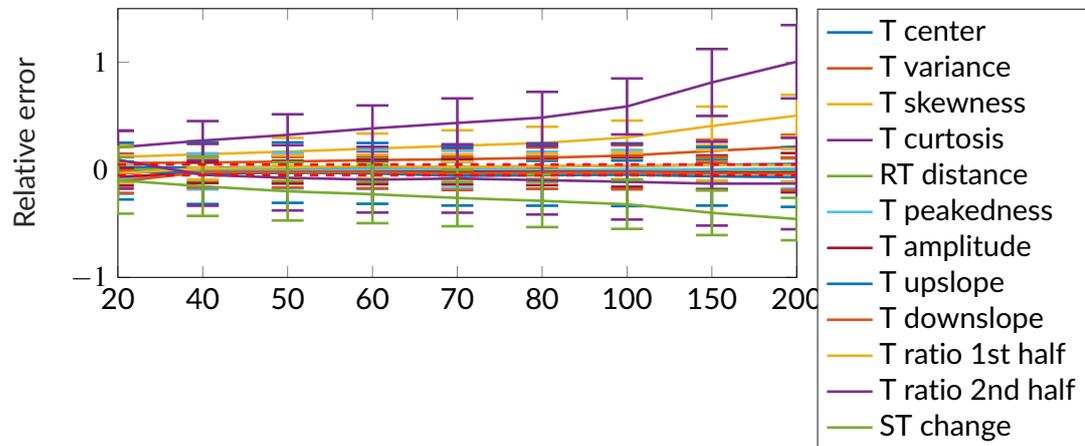


Figure A.1: Relative errors of T wave feature extraction algorithms for selected low-pass cut-off frequencies for a signal to noise ratio (SNR) of 0 dB. The algorithms were applied on the noisy filtered data and compared with the features extracted from the unfiltered noise-free data. The lines represent the median values of the relative errors, the bars visualize the interquartile ranges. HP: high-pass, LP: low-pass; all values on the x-axis are cut-off frequencies in Hz. The red dashed lines mark errors of $\pm 5\%$.

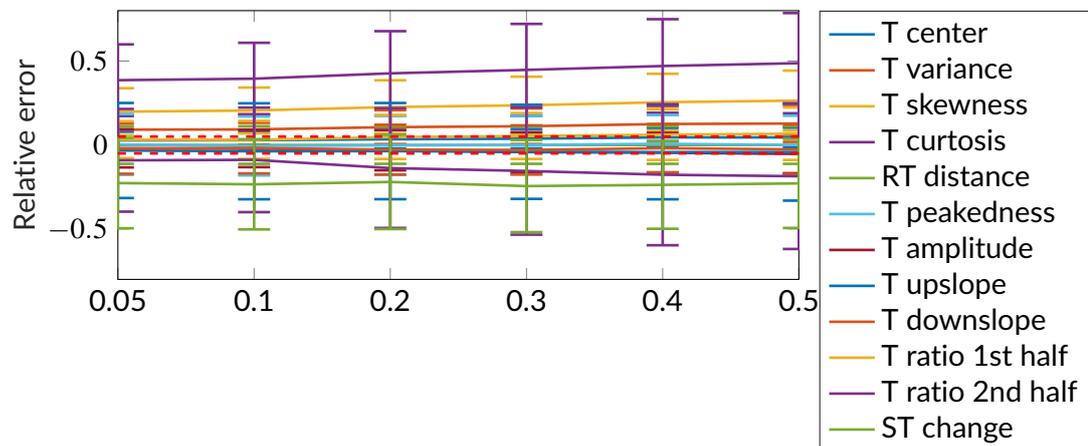


Figure A.2: Relative errors of T wave feature extraction algorithms for selected high-pass cut-off frequencies for an signal to noise ratio (SNR) of 0 dB. The algorithms were applied on the noisy filtered data and compared with the features extracted from the unfiltered noise-free data. The lines represent the median values of the relative errors, the bars visualize the interquartile ranges. HP: high-pass, LP: low-pass; all values on the x-axis are cut-off frequencies in Hz. The red dashed lines mark errors of $\pm 5\%$.

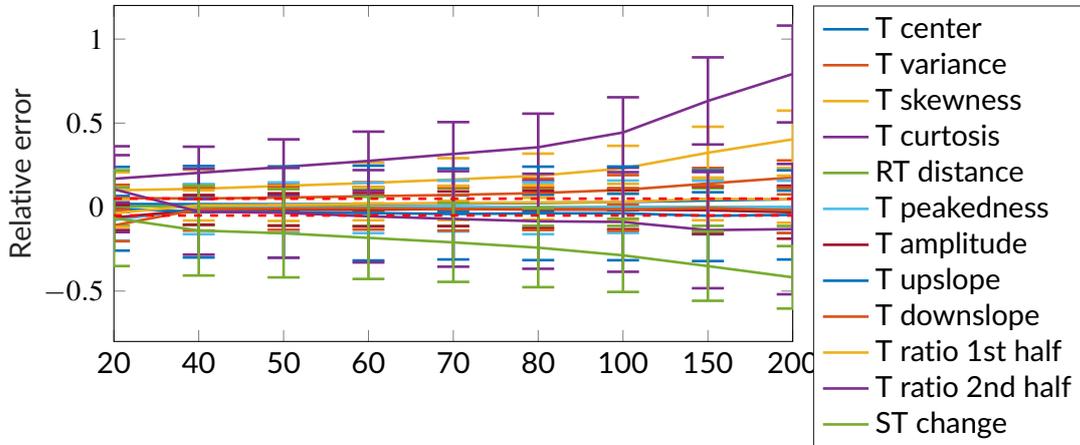


Figure A.3: Relative errors of T wave feature extraction algorithms for selected low-pass cut-off frequencies for an signal to noise ratio (SNR) of 3 dB. The algorithms were applied on the noisy filtered data and compared with the features extracted from the unfiltered noise-free data. The lines represent the median values of the relative errors, the bars visualize the interquartile ranges. HP: high-pass, LP: low-pass; all values on the x-axis are cut-off frequencies in Hz. The red dashed lines mark errors of $\pm 5\%$.

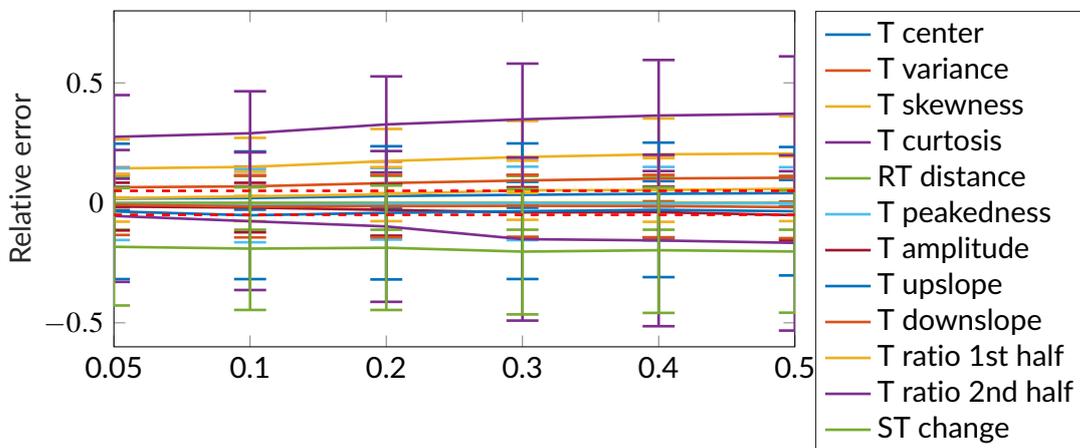


Figure A.4: Relative errors of T wave feature extraction algorithms for selected high-pass cut-off frequencies for an signal to noise ratio (SNR) of 3 dB. The algorithms were applied on the noisy filtered data and compared with the features extracted from the unfiltered noise-free data. The lines represent the median values of the relative errors, the bars visualize the interquartile ranges. HP: high-pass, LP: low-pass; all values on the x-axis are cut-off frequencies in Hz. The red dashed lines mark errors of $\pm 5\%$.

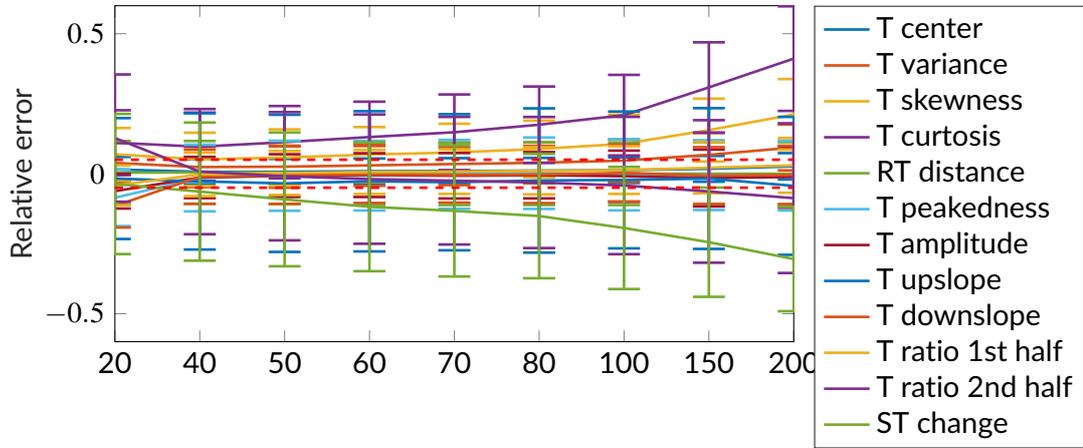


Figure A.5: Relative errors of T wave feature extraction algorithms for selected low-pass cut-off frequencies for an signal to noise ratio (SNR) of 10 dB. The algorithms were applied on the noisy filtered data and compared with the features extracted from the unfiltered noise-free data. The lines represent the median values of the relative errors, the bars visualize the interquartile ranges. HP: high-pass, LP: low-pass; all values on the x-axis are cut-off frequencies in Hz. The red dashed lines mark errors of $\pm 5\%$.

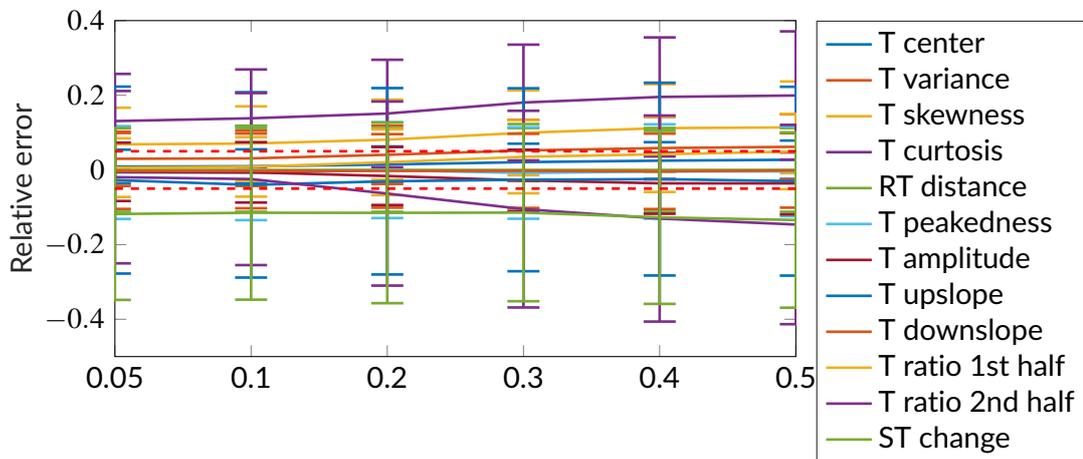


Figure A.6: Relative errors of T wave feature extraction algorithms for selected high-pass cut-off frequencies for an signal to noise ratio (SNR) of 10 dB. The algorithms were applied on the noisy filtered data and compared with the features extracted from the unfiltered noise-free data. The lines represent the median values of the relative errors, the bars visualize the interquartile ranges. HP: high-pass, LP: low-pass; all values on the x-axis are cut-off frequencies in Hz. The red dashed lines mark errors of $\pm 5\%$.

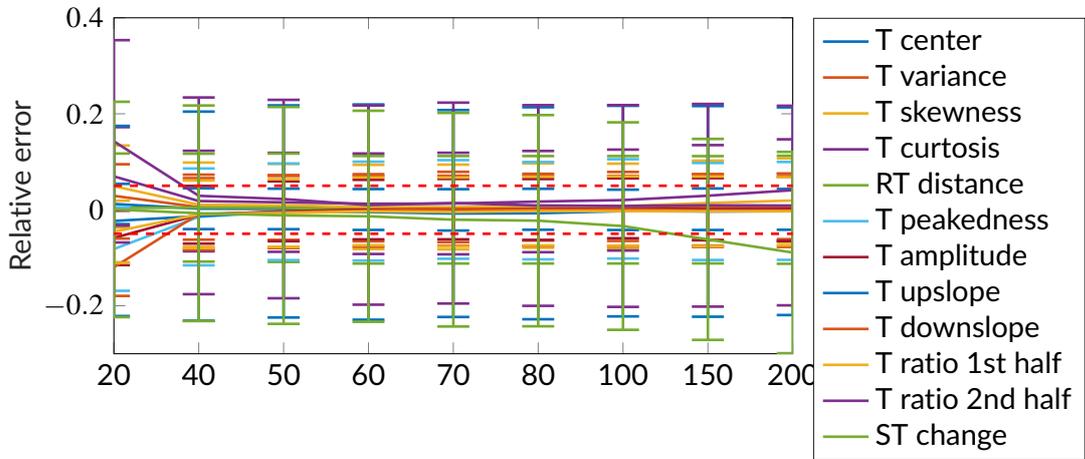


Figure A.7: Relative errors of T wave feature extraction algorithms for selected low-pass cut-off frequencies for an signal to noise ratio (SNR) of 30 dB. The algorithms were applied on the noisy filtered data and compared with the features extracted from the unfiltered noise-free data. The lines represent the median values of the relative errors, the bars visualize the interquartile ranges. HP: high-pass, LP: low-pass; all values on the x-axis are cut-off frequencies in Hz. The red dashed lines mark errors of $\pm 5\%$.

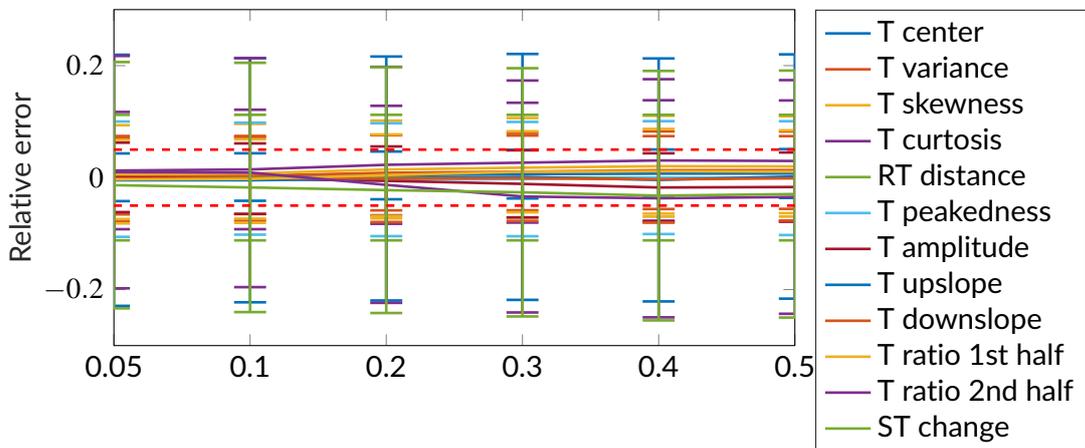


Figure A.8: Relative errors of T wave feature extraction algorithms for selected high-pass cut-off frequencies for an signal to noise ratio (SNR) of 30 dB. The algorithms were applied on the noisy filtered data and compared with the features extracted from the unfiltered noise-free data. The lines represent the median values of the relative errors, the bars visualize the interquartile ranges. HP: high-pass, LP: low-pass; all values on the x-axis are cut-off frequencies in Hz. The red dashed lines mark errors of $\pm 5\%$.

A.3 Influence of Patient Geometry

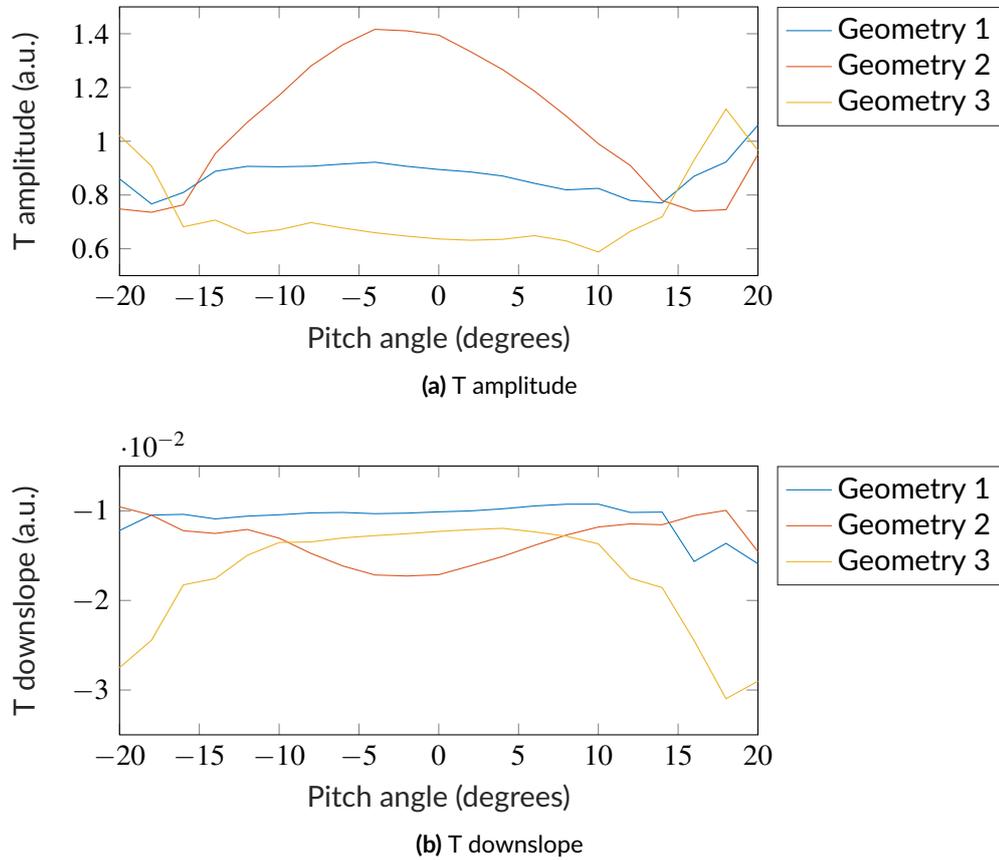


Figure A.9: Feature changes dependent on γ_{pitch} for the three geometries for the default concentration of the Himeno et al. model.

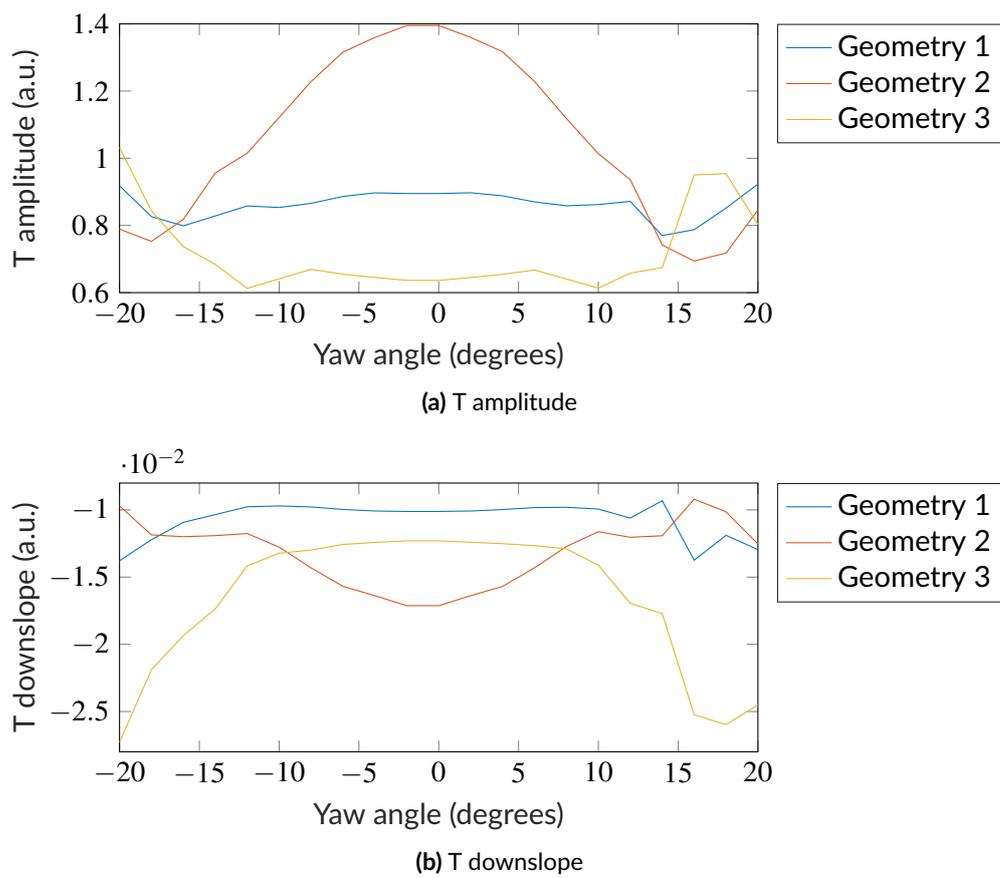


Figure A.10: Feature changes dependent on γ_{yaw} for the three geometries for the default concentration of the Himeno et al. model.

A.4 ECGdeli - ECG Delineation Algorithms

Table A.2: Detection errors of ECGdeli, ecg-kit and ecgpuwave compared to manual expert annotations in samples for those annotations that were found by all three methods. The number of detected points refers to the number of waves that were also annotated, so it can be maximum the number of annotations. med: median, iqr: interquartile range, m: mean, std: standard deviation, signed: signed errors, abs: absolute errors.

	P Onset	P Peak	P Offset
med±iqr ECGdeli	2.00±3.00	1.00±2.00	3.00±3.00
m±std abs ECGdeli	3.61±6.08	2.18±5.45	3.75±6.21
m±std signed ECGdeli	-1.54±6.90	0.65±5.83	2.21±6.91
med±iqr ECGKit	2.00±4.00	1.00±2.00	2.00±3.00
m±std abs ECGKit	2.91±4.99	2.43±4.83	2.96±3.66
m±std signed ECGKit	-1.10±5.68	-1.09±5.30	-2.13±4.20
med±iqr ecgpu	3.00±4.00	2.00±2.00	2.00±3.00
m±std abs ecgpu	4.26±5.22	2.74±3.90	3.75±4.63
m±std signed ecgpu	3.05±6.00	-0.50±4.75	-1.39±5.80
Analyzed annotations	2096	2096	2096
	QRS Onset	QRS Peak	QRS Offset
med±iqr ECGdeli	2.00±4.00	1.00±1.00	3.00±4.00
m±std abs ECGdeli	3.96±7.99	2.46±8.46	4.35±8.13
m±std signed ECGdeli	-1.36±8.81	-0.60±8.79	2.15±8.97
med±iqr ECGKit	1.00±4.00	2.00±2.00	1.00±3.00
m±std abs ECGKit	1.66±2.14	3.36±4.37	1.91±2.40
m±std signed ECGKit	0.54±2.65	-3.13±4.54	-0.46±3.04
med±iqr ecgpu	2.00±4.00	3.00±2.00	2.00±3.00
m±std abs ecgpu	3.46±3.55	3.54±4.32	3.50±5.47
m±std signed ecgpu	-1.61±4.69	-1.72±5.32	-0.38±6.48
Analyzed annotations	4018	4013	4016
	T Onset	T Peak	T Offset
med±iqr ECGdeli	5.00±9.00	2.00±3.00	3.00±6.00
m±std abs ECGdeli	8.36±9.17	5.78±11.80	6.79±10.84
m±std signed ECGdeli	-2.97±12.05	-0.19±13.14	-1.47±12.71
med±iqr ECGKit	5.00±14.00	1.00±14.00	2.00±11.00
m±std abs ECGKit	9.92±13.58	4.49±12.18	6.60±14.42
m±std signed ECGKit	-1.10±16.78	-2.17±12.80	-2.88±15.59
med±iqr ecgpu	8.00±14.00	2.00±14.00	5.00±11.00
m±std abs ecgpu	12.80±13.53	10.49±17.49	11.52±16.55
m±std signed ecgpu	9.65±15.94	7.45±18.98	3.90±19.78
Analyzed annotations	1237	3731	3741

A.5 Convolutional Neural Network Structures

A.5.1 ScaleNet

```

ResNet(
  (conv1): Conv2d(1, 64, kernel_size=(5, 5), stride=(2, 2), padding=(3,
    3), bias=False)
  (bn1): BatchNorm2d(64, eps=1e-05, momentum=0.1, affine=True,
    track_running_stats=True)
  (relu): ReLU(inplace=True)
  (maxpool): MaxPool2d(kernel_size=3, stride=2, padding=1, dilation=1,
    ceil_mode=False)
  (layer1): Sequential(
    (0): Bottleneck(
      (conv1): Conv2d(64, 128, kernel_size=(1, 1), stride=(1, 1), bias=
        False)
      (bn1): BatchNorm2d(128, eps=1e-05, momentum=0.1, affine=True,
        track_running_stats=True)
      (conv2): Conv2d(128, 128, kernel_size=(3, 3), stride=(1, 1),
        padding=(1, 1), groups=32, bias=False)
      (bn2): BatchNorm2d(128, eps=1e-05, momentum=0.1, affine=True,
        track_running_stats=True)
      (conv3): Conv2d(128, 256, kernel_size=(1, 1), stride=(1, 1), bias=
        False)
      (bn3): BatchNorm2d(256, eps=1e-05, momentum=0.1, affine=True,
        track_running_stats=True)
      (relu): ReLU(inplace=True)
      (downsample): Sequential(
        (0): Conv2d(64, 256, kernel_size=(1, 1), stride=(1, 1), bias=
          False)
        (1): BatchNorm2d(256, eps=1e-05, momentum=0.1, affine=True,
          track_running_stats=True)
      )
    )
  )
  (1): Bottleneck(
    (conv1): Conv2d(256, 128, kernel_size=(1, 1), stride=(1, 1), bias=
      False)
    (bn1): BatchNorm2d(128, eps=1e-05, momentum=0.1, affine=True,
      track_running_stats=True)
    (conv2): Conv2d(128, 128, kernel_size=(3, 3), stride=(1, 1),
      padding=(1, 1), groups=32, bias=False)
    (bn2): BatchNorm2d(128, eps=1e-05, momentum=0.1, affine=True,
      track_running_stats=True)
    (conv3): Conv2d(128, 256, kernel_size=(1, 1), stride=(1, 1), bias=
      False)
    (bn3): BatchNorm2d(256, eps=1e-05, momentum=0.1, affine=True,
      track_running_stats=True)
    (relu): ReLU(inplace=True)
  )
)

```

```
(2): Bottleneck(
  (conv1): Conv2d(256, 128, kernel_size=(1, 1), stride=(1, 1), bias=
    False)
  (bn1): BatchNorm2d(128, eps=1e-05, momentum=0.1, affine=True,
    track_running_stats=True)
  (conv2): Conv2d(128, 128, kernel_size=(3, 3), stride=(1, 1),
    padding=(1, 1), groups=32, bias=False)
  (bn2): BatchNorm2d(128, eps=1e-05, momentum=0.1, affine=True,
    track_running_stats=True)
  (conv3): Conv2d(128, 256, kernel_size=(1, 1), stride=(1, 1), bias=
    False)
  (bn3): BatchNorm2d(256, eps=1e-05, momentum=0.1, affine=True,
    track_running_stats=True)
  (relu): ReLU(inplace=True)
)
)
(layer2): Sequential(
  (0): Bottleneck(
    (conv1): Conv2d(256, 256, kernel_size=(1, 1), stride=(1, 1), bias=
      False)
    (bn1): BatchNorm2d(256, eps=1e-05, momentum=0.1, affine=True,
      track_running_stats=True)
    (conv2): Conv2d(256, 256, kernel_size=(3, 3), stride=(2, 2),
      padding=(1, 1), groups=32, bias=False)
    (bn2): BatchNorm2d(256, eps=1e-05, momentum=0.1, affine=True,
      track_running_stats=True)
    (conv3): Conv2d(256, 512, kernel_size=(1, 1), stride=(1, 1), bias=
      False)
    (bn3): BatchNorm2d(512, eps=1e-05, momentum=0.1, affine=True,
      track_running_stats=True)
    (relu): ReLU(inplace=True)
    (downsample): Sequential(
      (0): Conv2d(256, 512, kernel_size=(1, 1), stride=(2, 2), bias=
        False)
      (1): BatchNorm2d(512, eps=1e-05, momentum=0.1, affine=True,
        track_running_stats=True)
    )
  )
)
  (1): Bottleneck(
    (conv1): Conv2d(512, 256, kernel_size=(1, 1), stride=(1, 1), bias=
      False)
    (bn1): BatchNorm2d(256, eps=1e-05, momentum=0.1, affine=True,
      track_running_stats=True)
    (conv2): Conv2d(256, 256, kernel_size=(3, 3), stride=(1, 1),
      padding=(1, 1), groups=32, bias=False)
    (bn2): BatchNorm2d(256, eps=1e-05, momentum=0.1, affine=True,
      track_running_stats=True)
    (conv3): Conv2d(256, 512, kernel_size=(1, 1), stride=(1, 1), bias=
      False)
    (bn3): BatchNorm2d(512, eps=1e-05, momentum=0.1, affine=True,
      track_running_stats=True)
    (relu): ReLU(inplace=True)
```

```

)
(2): Bottleneck(
  (conv1): Conv2d(512, 256, kernel_size=(1, 1), stride=(1, 1), bias=
    False)
  (bn1): BatchNorm2d(256, eps=1e-05, momentum=0.1, affine=True,
    track_running_stats=True)
  (conv2): Conv2d(256, 256, kernel_size=(3, 3), stride=(1, 1),
    padding=(1, 1), groups=32, bias=False)
  (bn2): BatchNorm2d(256, eps=1e-05, momentum=0.1, affine=True,
    track_running_stats=True)
  (conv3): Conv2d(256, 512, kernel_size=(1, 1), stride=(1, 1), bias=
    False)
  (bn3): BatchNorm2d(512, eps=1e-05, momentum=0.1, affine=True,
    track_running_stats=True)
  (relu): ReLU(inplace=True)
)
(3): Bottleneck(
  (conv1): Conv2d(512, 256, kernel_size=(1, 1), stride=(1, 1), bias=
    False)
  (bn1): BatchNorm2d(256, eps=1e-05, momentum=0.1, affine=True,
    track_running_stats=True)
  (conv2): Conv2d(256, 256, kernel_size=(3, 3), stride=(1, 1),
    padding=(1, 1), groups=32, bias=False)
  (bn2): BatchNorm2d(256, eps=1e-05, momentum=0.1, affine=True,
    track_running_stats=True)
  (conv3): Conv2d(256, 512, kernel_size=(1, 1), stride=(1, 1), bias=
    False)
  (bn3): BatchNorm2d(512, eps=1e-05, momentum=0.1, affine=True,
    track_running_stats=True)
  (relu): ReLU(inplace=True)
)
)
(layer3): Sequential(
  (0): Bottleneck(
    (conv1): Conv2d(512, 512, kernel_size=(1, 1), stride=(1, 1), bias=
      False)
    (bn1): BatchNorm2d(512, eps=1e-05, momentum=0.1, affine=True,
      track_running_stats=True)
    (conv2): Conv2d(512, 512, kernel_size=(3, 3), stride=(2, 2),
      padding=(1, 1), groups=32, bias=False)
    (bn2): BatchNorm2d(512, eps=1e-05, momentum=0.1, affine=True,
      track_running_stats=True)
    (conv3): Conv2d(512, 1024, kernel_size=(1, 1), stride=(1, 1), bias=
      False)
    (bn3): BatchNorm2d(1024, eps=1e-05, momentum=0.1, affine=True,
      track_running_stats=True)
    (relu): ReLU(inplace=True)
  )
  (downsample): Sequential(
    (0): Conv2d(512, 1024, kernel_size=(1, 1), stride=(2, 2), bias=
      False)
    (1): BatchNorm2d(1024, eps=1e-05, momentum=0.1, affine=True,
      track_running_stats=True)
  )
)

```

```
)
)
(1): Bottleneck(
  (conv1): Conv2d(1024, 512, kernel_size=(1, 1), stride=(1, 1), bias=
    False)
  (bn1): BatchNorm2d(512, eps=1e-05, momentum=0.1, affine=True,
    track_running_stats=True)
  (conv2): Conv2d(512, 512, kernel_size=(3, 3), stride=(1, 1),
    padding=(1, 1), groups=32, bias=False)
  (bn2): BatchNorm2d(512, eps=1e-05, momentum=0.1, affine=True,
    track_running_stats=True)
  (conv3): Conv2d(512, 1024, kernel_size=(1, 1), stride=(1, 1), bias=
    False)
  (bn3): BatchNorm2d(1024, eps=1e-05, momentum=0.1, affine=True,
    track_running_stats=True)
  (relu): ReLU(inplace=True)
)
(2): Bottleneck(
  (conv1): Conv2d(1024, 512, kernel_size=(1, 1), stride=(1, 1), bias=
    False)
  (bn1): BatchNorm2d(512, eps=1e-05, momentum=0.1, affine=True,
    track_running_stats=True)
  (conv2): Conv2d(512, 512, kernel_size=(3, 3), stride=(1, 1),
    padding=(1, 1), groups=32, bias=False)
  (bn2): BatchNorm2d(512, eps=1e-05, momentum=0.1, affine=True,
    track_running_stats=True)
  (conv3): Conv2d(512, 1024, kernel_size=(1, 1), stride=(1, 1), bias=
    False)
  (bn3): BatchNorm2d(1024, eps=1e-05, momentum=0.1, affine=True,
    track_running_stats=True)
  (relu): ReLU(inplace=True)
)
(3): Bottleneck(
  (conv1): Conv2d(1024, 512, kernel_size=(1, 1), stride=(1, 1), bias=
    False)
  (bn1): BatchNorm2d(512, eps=1e-05, momentum=0.1, affine=True,
    track_running_stats=True)
  (conv2): Conv2d(512, 512, kernel_size=(3, 3), stride=(1, 1),
    padding=(1, 1), groups=32, bias=False)
  (bn2): BatchNorm2d(512, eps=1e-05, momentum=0.1, affine=True,
    track_running_stats=True)
  (conv3): Conv2d(512, 1024, kernel_size=(1, 1), stride=(1, 1), bias=
    False)
  (bn3): BatchNorm2d(1024, eps=1e-05, momentum=0.1, affine=True,
    track_running_stats=True)
  (relu): ReLU(inplace=True)
)
(4): Bottleneck(
  (conv1): Conv2d(1024, 512, kernel_size=(1, 1), stride=(1, 1), bias=
    False)
  (bn1): BatchNorm2d(512, eps=1e-05, momentum=0.1, affine=True,
    track_running_stats=True)
```

```

(conv2): Conv2d(512, 512, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), groups=32, bias=False)
.bn2): BatchNorm2d(512, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
(conv3): Conv2d(512, 1024, kernel_size=(1, 1), stride=(1, 1), bias=
False)
.bn3): BatchNorm2d(1024, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
(relu): ReLU(inplace=True)
)
(5): Bottleneck(
(conv1): Conv2d(1024, 512, kernel_size=(1, 1), stride=(1, 1), bias=
False)
.bn1): BatchNorm2d(512, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
(conv2): Conv2d(512, 512, kernel_size=(3, 3), stride=(1, 1),
padding=(1, 1), groups=32, bias=False)
.bn2): BatchNorm2d(512, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
(conv3): Conv2d(512, 1024, kernel_size=(1, 1), stride=(1, 1), bias=
False)
.bn3): BatchNorm2d(1024, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
(relu): ReLU(inplace=True)
)
)
(layer4): Sequential(
(0): Bottleneck(
(conv1): Conv2d(1024, 1024, kernel_size=(1, 1), stride=(1, 1), bias
=False)
.bn1): BatchNorm2d(1024, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
(conv2): Conv2d(1024, 1024, kernel_size=(3, 3), stride=(2, 2),
padding=(1, 1), groups=32, bias=False)
.bn2): BatchNorm2d(1024, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
(conv3): Conv2d(1024, 2048, kernel_size=(1, 1), stride=(1, 1), bias
=False)
.bn3): BatchNorm2d(2048, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
(relu): ReLU(inplace=True)
(downsample): Sequential(
(0): Conv2d(1024, 2048, kernel_size=(1, 1), stride=(2, 2), bias=
False)
(1): BatchNorm2d(2048, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
)
)
(1): Bottleneck(
(conv1): Conv2d(2048, 1024, kernel_size=(1, 1), stride=(1, 1), bias
=False)

```

```

    (bn1): BatchNorm2d(1024, eps=1e-05, momentum=0.1, affine=True,
      track_running_stats=True)
    (conv2): Conv2d(1024, 1024, kernel_size=(3, 3), stride=(1, 1),
      padding=(1, 1), groups=32, bias=False)
    (bn2): BatchNorm2d(1024, eps=1e-05, momentum=0.1, affine=True,
      track_running_stats=True)
    (conv3): Conv2d(1024, 2048, kernel_size=(1, 1), stride=(1, 1), bias
      =False)
    (bn3): BatchNorm2d(2048, eps=1e-05, momentum=0.1, affine=True,
      track_running_stats=True)
    (relu): ReLU(inplace=True)
  )
(2): Bottleneck(
  (conv1): Conv2d(2048, 1024, kernel_size=(1, 1), stride=(1, 1), bias
    =False)
  (bn1): BatchNorm2d(1024, eps=1e-05, momentum=0.1, affine=True,
    track_running_stats=True)
  (conv2): Conv2d(1024, 1024, kernel_size=(3, 3), stride=(1, 1),
    padding=(1, 1), groups=32, bias=False)
  (bn2): BatchNorm2d(1024, eps=1e-05, momentum=0.1, affine=True,
    track_running_stats=True)
  (conv3): Conv2d(1024, 2048, kernel_size=(1, 1), stride=(1, 1), bias
    =False)
  (bn3): BatchNorm2d(2048, eps=1e-05, momentum=0.1, affine=True,
    track_running_stats=True)
  (relu): ReLU(inplace=True)
)
)
(avgpool): AdaptiveAvgPool2d(output_size=(1, 1))
(fc): Sequential(
  (0): Linear(in_features=2048, out_features=2, bias=True)
  (1): Linear(in_features=2, out_features=2, bias=True)
)
)

```

Layer (type)	Output Shape	Param #
Conv2d-1	[-1, 64, 113, 176]	1,600
BatchNorm2d-2	[-1, 64, 113, 176]	128
ReLU-3	[-1, 64, 113, 176]	0
MaxPool2d-4	[-1, 64, 57, 88]	0
Conv2d-5	[-1, 128, 57, 88]	8,192
BatchNorm2d-6	[-1, 128, 57, 88]	256
ReLU-7	[-1, 128, 57, 88]	0
Conv2d-8	[-1, 128, 57, 88]	4,608
BatchNorm2d-9	[-1, 128, 57, 88]	256
ReLU-10	[-1, 128, 57, 88]	0
Conv2d-11	[-1, 256, 57, 88]	32,768
BatchNorm2d-12	[-1, 256, 57, 88]	512
Conv2d-13	[-1, 256, 57, 88]	16,384
BatchNorm2d-14	[-1, 256, 57, 88]	512

ReLU-15	[-1, 256, 57, 88]	0
Bottleneck-16	[-1, 256, 57, 88]	0
Conv2d-17	[-1, 128, 57, 88]	32,768
BatchNorm2d-18	[-1, 128, 57, 88]	256
ReLU-19	[-1, 128, 57, 88]	0
Conv2d-20	[-1, 128, 57, 88]	4,608
BatchNorm2d-21	[-1, 128, 57, 88]	256
ReLU-22	[-1, 128, 57, 88]	0
Conv2d-23	[-1, 256, 57, 88]	32,768
BatchNorm2d-24	[-1, 256, 57, 88]	512
ReLU-25	[-1, 256, 57, 88]	0
Bottleneck-26	[-1, 256, 57, 88]	0
Conv2d-27	[-1, 128, 57, 88]	32,768
BatchNorm2d-28	[-1, 128, 57, 88]	256
ReLU-29	[-1, 128, 57, 88]	0
Conv2d-30	[-1, 128, 57, 88]	4,608
BatchNorm2d-31	[-1, 128, 57, 88]	256
ReLU-32	[-1, 128, 57, 88]	0
Conv2d-33	[-1, 256, 57, 88]	32,768
BatchNorm2d-34	[-1, 256, 57, 88]	512
ReLU-35	[-1, 256, 57, 88]	0
Bottleneck-36	[-1, 256, 57, 88]	0
Conv2d-37	[-1, 256, 57, 88]	65,536
BatchNorm2d-38	[-1, 256, 57, 88]	512
ReLU-39	[-1, 256, 57, 88]	0
Conv2d-40	[-1, 256, 29, 44]	18,432
BatchNorm2d-41	[-1, 256, 29, 44]	512
ReLU-42	[-1, 256, 29, 44]	0
Conv2d-43	[-1, 512, 29, 44]	131,072
BatchNorm2d-44	[-1, 512, 29, 44]	1,024
Conv2d-45	[-1, 512, 29, 44]	131,072
BatchNorm2d-46	[-1, 512, 29, 44]	1,024
ReLU-47	[-1, 512, 29, 44]	0
Bottleneck-48	[-1, 512, 29, 44]	0
Conv2d-49	[-1, 256, 29, 44]	131,072
BatchNorm2d-50	[-1, 256, 29, 44]	512
ReLU-51	[-1, 256, 29, 44]	0
Conv2d-52	[-1, 256, 29, 44]	18,432
BatchNorm2d-53	[-1, 256, 29, 44]	512
ReLU-54	[-1, 256, 29, 44]	0
Conv2d-55	[-1, 512, 29, 44]	131,072
BatchNorm2d-56	[-1, 512, 29, 44]	1,024
ReLU-57	[-1, 512, 29, 44]	0
Bottleneck-58	[-1, 512, 29, 44]	0
Conv2d-59	[-1, 256, 29, 44]	131,072
BatchNorm2d-60	[-1, 256, 29, 44]	512
ReLU-61	[-1, 256, 29, 44]	0
Conv2d-62	[-1, 256, 29, 44]	18,432
BatchNorm2d-63	[-1, 256, 29, 44]	512
ReLU-64	[-1, 256, 29, 44]	0
Conv2d-65	[-1, 512, 29, 44]	131,072
BatchNorm2d-66	[-1, 512, 29, 44]	1,024

ReLU-67	[-1, 512, 29, 44]	0
Bottleneck -68	[-1, 512, 29, 44]	0
Conv2d-69	[-1, 256, 29, 44]	131,072
BatchNorm2d-70	[-1, 256, 29, 44]	512
ReLU-71	[-1, 256, 29, 44]	0
Conv2d-72	[-1, 256, 29, 44]	18,432
BatchNorm2d-73	[-1, 256, 29, 44]	512
ReLU-74	[-1, 256, 29, 44]	0
Conv2d-75	[-1, 512, 29, 44]	131,072
BatchNorm2d-76	[-1, 512, 29, 44]	1,024
ReLU-77	[-1, 512, 29, 44]	0
Bottleneck -78	[-1, 512, 29, 44]	0
Conv2d-79	[-1, 512, 29, 44]	262,144
BatchNorm2d-80	[-1, 512, 29, 44]	1,024
ReLU-81	[-1, 512, 29, 44]	0
Conv2d-82	[-1, 512, 15, 22]	73,728
BatchNorm2d-83	[-1, 512, 15, 22]	1,024
ReLU-84	[-1, 512, 15, 22]	0
Conv2d-85	[-1, 1024, 15, 22]	524,288
BatchNorm2d-86	[-1, 1024, 15, 22]	2,048
Conv2d-87	[-1, 1024, 15, 22]	524,288
BatchNorm2d-88	[-1, 1024, 15, 22]	2,048
ReLU-89	[-1, 1024, 15, 22]	0
Bottleneck -90	[-1, 1024, 15, 22]	0
Conv2d-91	[-1, 512, 15, 22]	524,288
BatchNorm2d-92	[-1, 512, 15, 22]	1,024
ReLU-93	[-1, 512, 15, 22]	0
Conv2d-94	[-1, 512, 15, 22]	73,728
BatchNorm2d-95	[-1, 512, 15, 22]	1,024
ReLU-96	[-1, 512, 15, 22]	0
Conv2d-97	[-1, 1024, 15, 22]	524,288
BatchNorm2d-98	[-1, 1024, 15, 22]	2,048
ReLU-99	[-1, 1024, 15, 22]	0
Bottleneck -100	[-1, 1024, 15, 22]	0
Conv2d-101	[-1, 512, 15, 22]	524,288
BatchNorm2d-102	[-1, 512, 15, 22]	1,024
ReLU-103	[-1, 512, 15, 22]	0
Conv2d-104	[-1, 512, 15, 22]	73,728
BatchNorm2d-105	[-1, 512, 15, 22]	1,024
ReLU-106	[-1, 512, 15, 22]	0
Conv2d-107	[-1, 1024, 15, 22]	524,288
BatchNorm2d-108	[-1, 1024, 15, 22]	2,048
ReLU-109	[-1, 1024, 15, 22]	0
Bottleneck -110	[-1, 1024, 15, 22]	0
Conv2d-111	[-1, 512, 15, 22]	524,288
BatchNorm2d-112	[-1, 512, 15, 22]	1,024
ReLU-113	[-1, 512, 15, 22]	0
Conv2d-114	[-1, 512, 15, 22]	73,728
BatchNorm2d-115	[-1, 512, 15, 22]	1,024
ReLU-116	[-1, 512, 15, 22]	0
Conv2d-117	[-1, 1024, 15, 22]	524,288
BatchNorm2d-118	[-1, 1024, 15, 22]	2,048

ReLU-119	[-1, 1024, 15, 22]	0
Bottleneck -120	[-1, 1024, 15, 22]	0
Conv2d-121	[-1, 512, 15, 22]	524,288
BatchNorm2d-122	[-1, 512, 15, 22]	1,024
ReLU-123	[-1, 512, 15, 22]	0
Conv2d-124	[-1, 512, 15, 22]	73,728
BatchNorm2d-125	[-1, 512, 15, 22]	1,024
ReLU-126	[-1, 512, 15, 22]	0
Conv2d-127	[-1, 1024, 15, 22]	524,288
BatchNorm2d-128	[-1, 1024, 15, 22]	2,048
ReLU-129	[-1, 1024, 15, 22]	0
Bottleneck -130	[-1, 1024, 15, 22]	0
Conv2d-131	[-1, 512, 15, 22]	524,288
BatchNorm2d-132	[-1, 512, 15, 22]	1,024
ReLU-133	[-1, 512, 15, 22]	0
Conv2d-134	[-1, 512, 15, 22]	73,728
BatchNorm2d-135	[-1, 512, 15, 22]	1,024
ReLU-136	[-1, 512, 15, 22]	0
Conv2d-137	[-1, 1024, 15, 22]	524,288
BatchNorm2d-138	[-1, 1024, 15, 22]	2,048
ReLU-139	[-1, 1024, 15, 22]	0
Bottleneck -140	[-1, 1024, 15, 22]	0
Conv2d-141	[-1, 1024, 15, 22]	1,048,576
BatchNorm2d-142	[-1, 1024, 15, 22]	2,048
ReLU-143	[-1, 1024, 15, 22]	0
Conv2d-144	[-1, 1024, 8, 11]	294,912
BatchNorm2d-145	[-1, 1024, 8, 11]	2,048
ReLU-146	[-1, 1024, 8, 11]	0
Conv2d-147	[-1, 2048, 8, 11]	2,097,152
BatchNorm2d-148	[-1, 2048, 8, 11]	4,096
Conv2d-149	[-1, 2048, 8, 11]	2,097,152
BatchNorm2d-150	[-1, 2048, 8, 11]	4,096
ReLU-151	[-1, 2048, 8, 11]	0
Bottleneck -152	[-1, 2048, 8, 11]	0
Conv2d-153	[-1, 1024, 8, 11]	2,097,152
BatchNorm2d-154	[-1, 1024, 8, 11]	2,048
ReLU-155	[-1, 1024, 8, 11]	0
Conv2d-156	[-1, 1024, 8, 11]	294,912
BatchNorm2d-157	[-1, 1024, 8, 11]	2,048
ReLU-158	[-1, 1024, 8, 11]	0
Conv2d-159	[-1, 2048, 8, 11]	2,097,152
BatchNorm2d-160	[-1, 2048, 8, 11]	4,096
ReLU-161	[-1, 2048, 8, 11]	0
Bottleneck -162	[-1, 2048, 8, 11]	0
Conv2d-163	[-1, 1024, 8, 11]	2,097,152
BatchNorm2d-164	[-1, 1024, 8, 11]	2,048
ReLU-165	[-1, 1024, 8, 11]	0
Conv2d-166	[-1, 1024, 8, 11]	294,912
BatchNorm2d-167	[-1, 1024, 8, 11]	2,048
ReLU-168	[-1, 1024, 8, 11]	0
Conv2d-169	[-1, 2048, 8, 11]	2,097,152
BatchNorm2d-170	[-1, 2048, 8, 11]	4,096

ReLU-171	[-1, 2048, 8, 11]	0
Bottleneck-172	[-1, 2048, 8, 11]	0
AdaptiveAvgPool2d-173	[-1, 2048, 1, 1]	0
Linear-174	[-1, 2]	4,098
Linear-175	[-1, 2]	6
=====		
Total params:	22,976,200	
Trainable params:	22,976,200	
Non-trainable params:	0	

Input size (MB):	0.30	
Forward/backward pass size (MB):	590.19	
Params size (MB):	87.65	
Estimated Total Size (MB):	678.14	

A.5.2 LocaNets

The fully connected layers from ScaleNet were changed from

```
...
(avgpool): AdaptiveAvgPool2d(output_size=(1, 1))
(fc): Sequential(
  (0): Linear(in_features=2048, out_features=2, bias=True)
  (1): Linear(in_features=2, out_features=2, bias=True)
)
```

for FuzzyNet93 to:

```
...
(avgpool): AdaptiveAvgPool2d(output_size=(1, 1))
(fc): Sequential(
  (0): Dropout(p=0.2, inplace=False)
  (1): Linear(in_features=2048, out_features=2048, bias=True)
  (2): Dropout(p=0.2, inplace=False)
  (3): Linear(in_features=2048, out_features=93, bias=True)
)
```

for FuzzyNet303 to:

```
...
(avgpool): AdaptiveAvgPool2d(output_size=(1, 1))
(fc): Sequential(
  (0): Dropout(p=0.2, inplace=False)
  (1): Linear(in_features=2048, out_features=2048, bias=True)
  (2): Dropout(p=0.2, inplace=False)
  (3): Linear(in_features=2048, out_features=303, bias=True)
)
```

and for RegressionNet to:

```
...
(avgpool): AdaptiveAvgPool2d(output_size=(1, 1))
```

```
(fc): Sequential(
  (0): Dropout(p=0.2, inplace=False)
  (1): Linear(in_features=2048, out_features=2048, bias=True)
  (2): Dropout(p=0.2, inplace=False)
  (3): Linear(in_features=2048, out_features=5, bias=True)
)
```

The structure and parameter listings as shown for ScaleNet change from

Layer (type)	Output Shape	Param #
Conv2d-1	[-1, 64, 113, 176]	1,600
...		
...		
AdaptiveAvgPool2d-173	[-1, 2048, 1, 1]	0
Linear-174	[-1, 2]	4,098
Linear-175	[-1, 2]	6
=====		
Total params: 22,976,200		
Trainable params: 22,976,200		
Non-trainable params: 0		
=====		
Input size (MB): 0.30		
Forward/backward pass size (MB): 590.19		
Params size (MB): 87.65		
Estimated Total Size (MB): 678.14		
=====		

for FuzzyNet93 to:

Layer (type)	Output Shape	Param #
Conv2d-1	[-1, 64, 113, 64]	1,600
...		
...		
AdaptiveAvgPool2d-173	[-1, 2048, 1, 1]	0
Dropout-174	[-1, 2048]	0
Linear-175	[-1, 2048]	4,196,352
Dropout-176	[-1, 2048]	0
Linear-177	[-1, 93]	190,557
=====		
Total params: 27,359,005		
Trainable params: 27,359,005		
Non-trainable params: 0		
=====		
Input size (MB): 0.11		
Forward/backward pass size (MB): 214.67		
Params size (MB): 104.37		
Estimated Total Size (MB): 319.15		
=====		

for FuzzyNet303 to:

AdaptiveAvgPool2d-173	[-1, 2048, 1, 1]	0
Dropout-174	[-1, 2048]	0
Linear-175	[-1, 2048]	4,196,352
Dropout-176	[-1, 2048]	0
Linear-177	[-1, 303]	620,847

```
=====
Total params: 27,789,295
Trainable params: 27,789,295
Non-trainable params: 0
-----
```

```
Input size (MB): 0.11
Forward/backward pass size (MB): 214.67
Params size (MB): 106.01
Estimated Total Size (MB): 320.79
-----
```

and for RegressionNet to:

Layer (type)	Output Shape	Param #
Conv2d-1	[-1, 64, 113, 64]	1,600
...		
...		
AdaptiveAvgPool2d-173	[-1, 2048, 1, 1]	0
Dropout-174	[-1, 2048]	0
Linear-175	[-1, 2048]	4,196,352
Dropout-176	[-1, 2048]	0
Linear-177	[-1, 5]	10,245

```
=====
Total params: 27,178,693
Trainable params: 27,178,693
Non-trainable params: 0
-----
```

```
Input size (MB): 0.11
Forward/backward pass size (MB): 214.67
Params size (MB): 103.68
Estimated Total Size (MB): 318.46
-----
```

A.6 ScaleNet Error Histograms

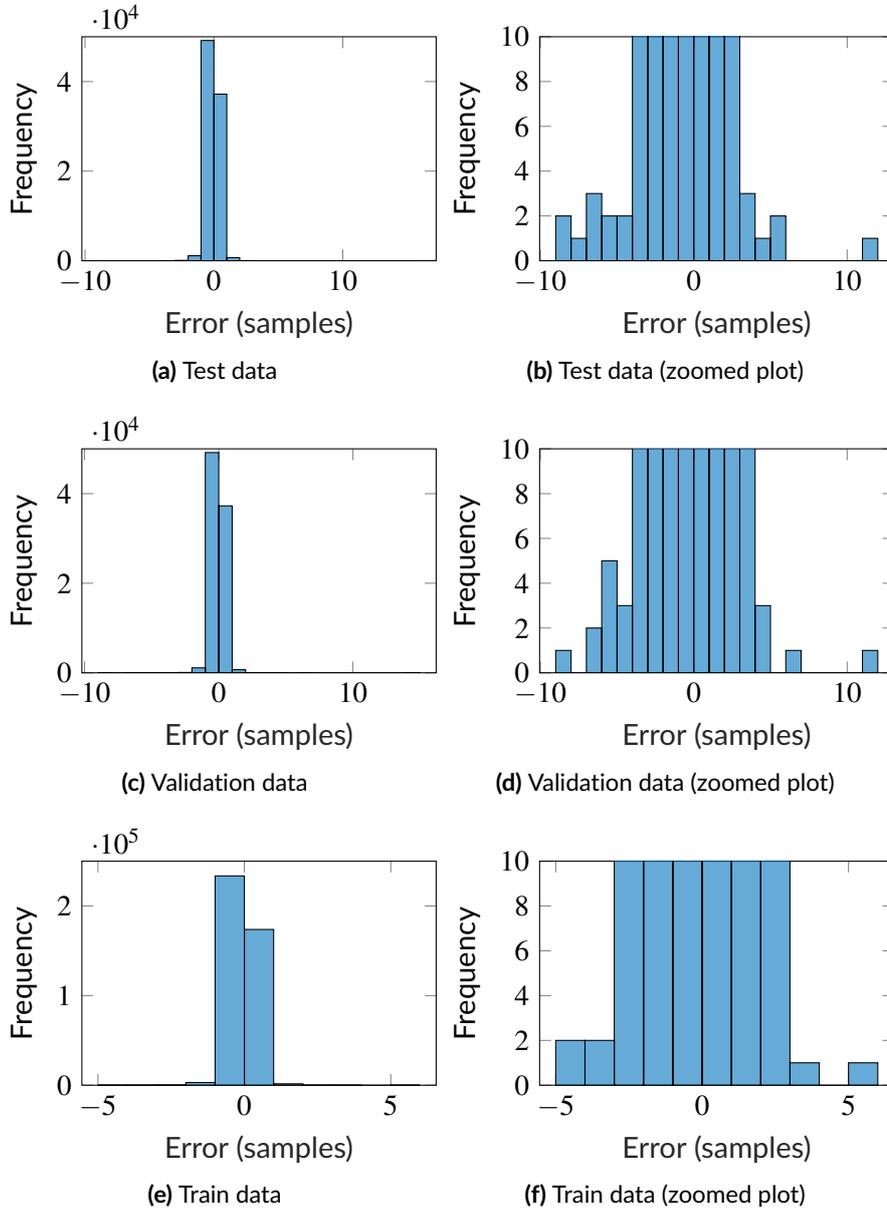


Figure A.11: Error histograms for the detection of the depolarization start with ScaleNet. The histograms are given for each data partition.

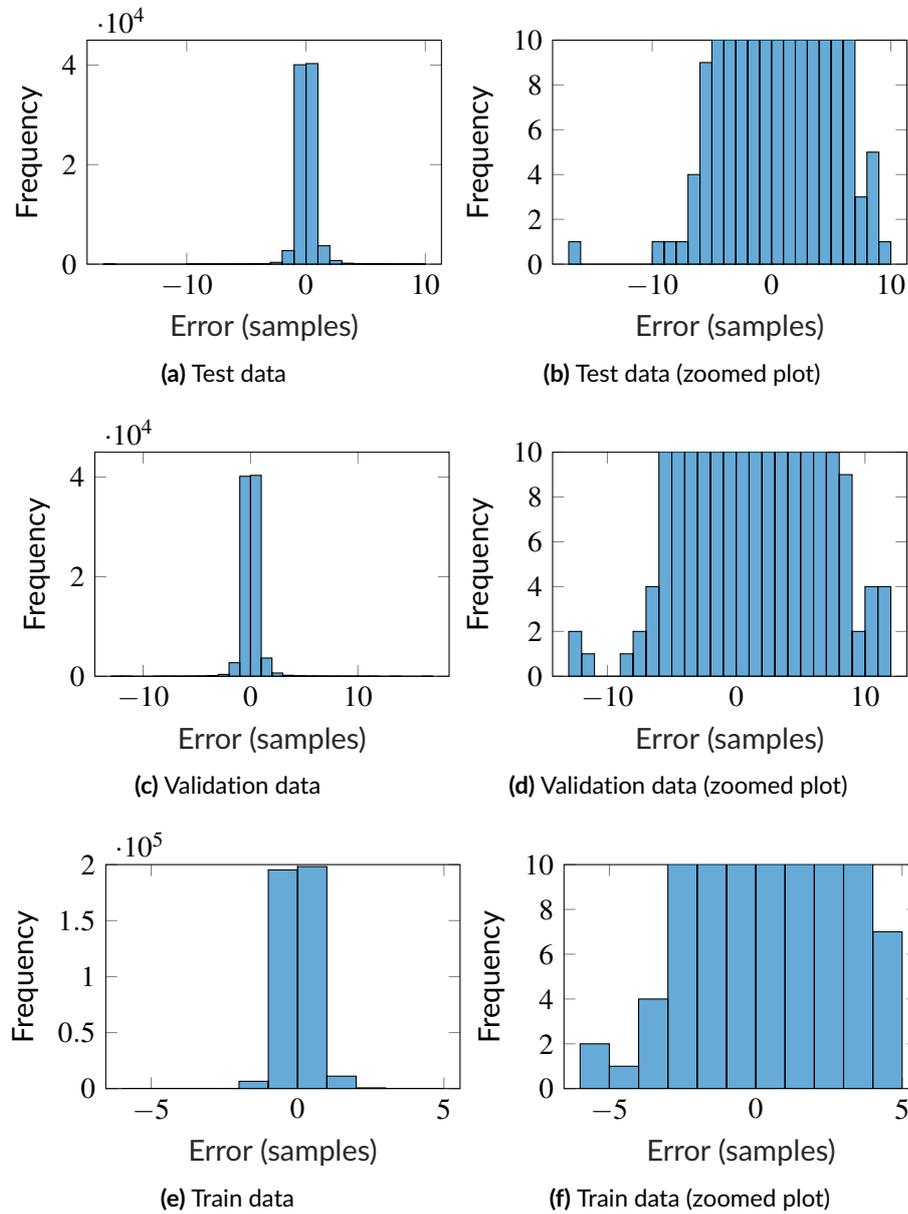


Figure A.12: Error histograms for the detection of the depolarization end with ScaleNet. The histograms are given for each data partition.

A.7 LocaNet Error Histograms

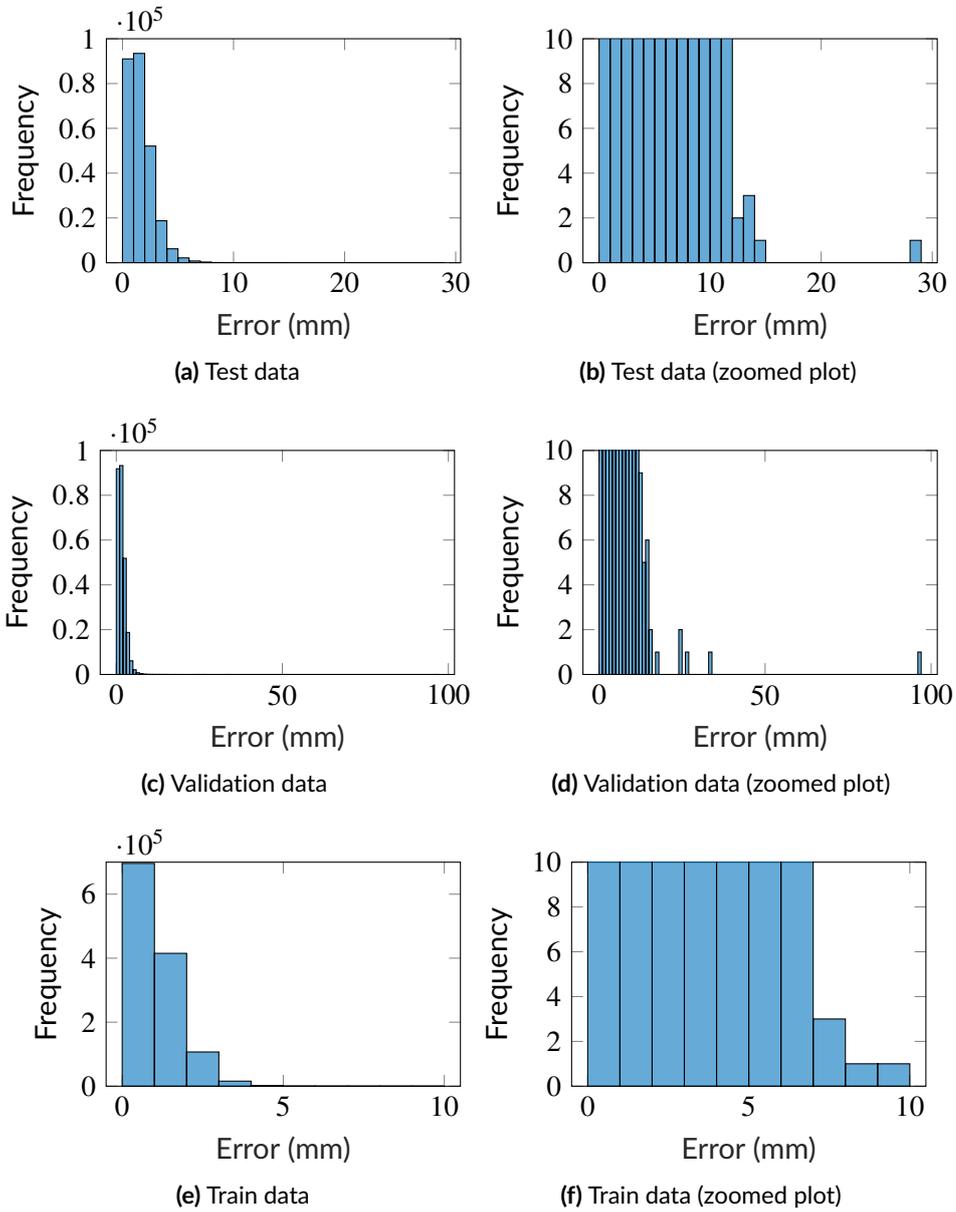


Figure A.13: Histogram of the localization error for FuzzyNet303. The histograms are given for each data partition.

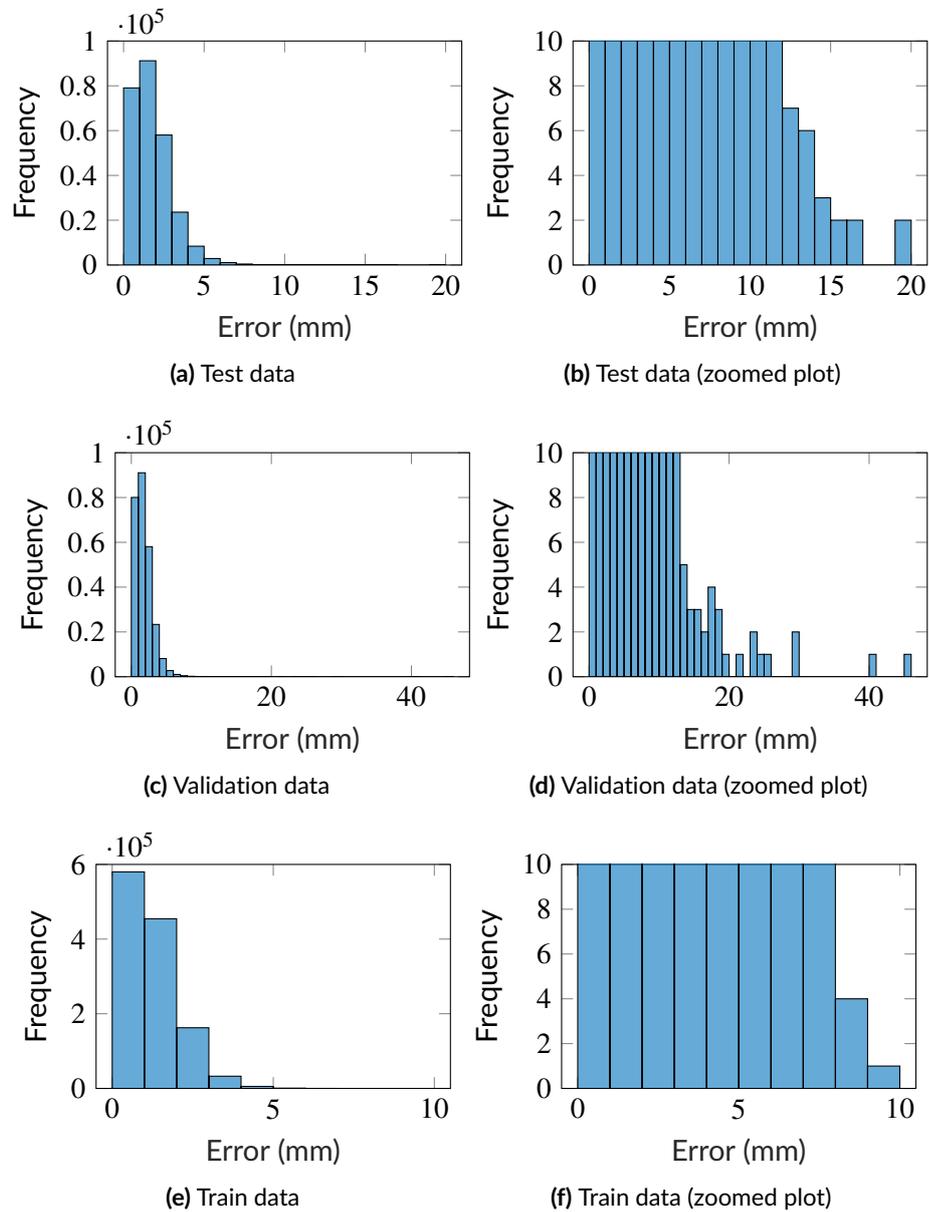


Figure A.14: Histogram of the localization error for FuzzyNet93. The histograms are given for each data partition.

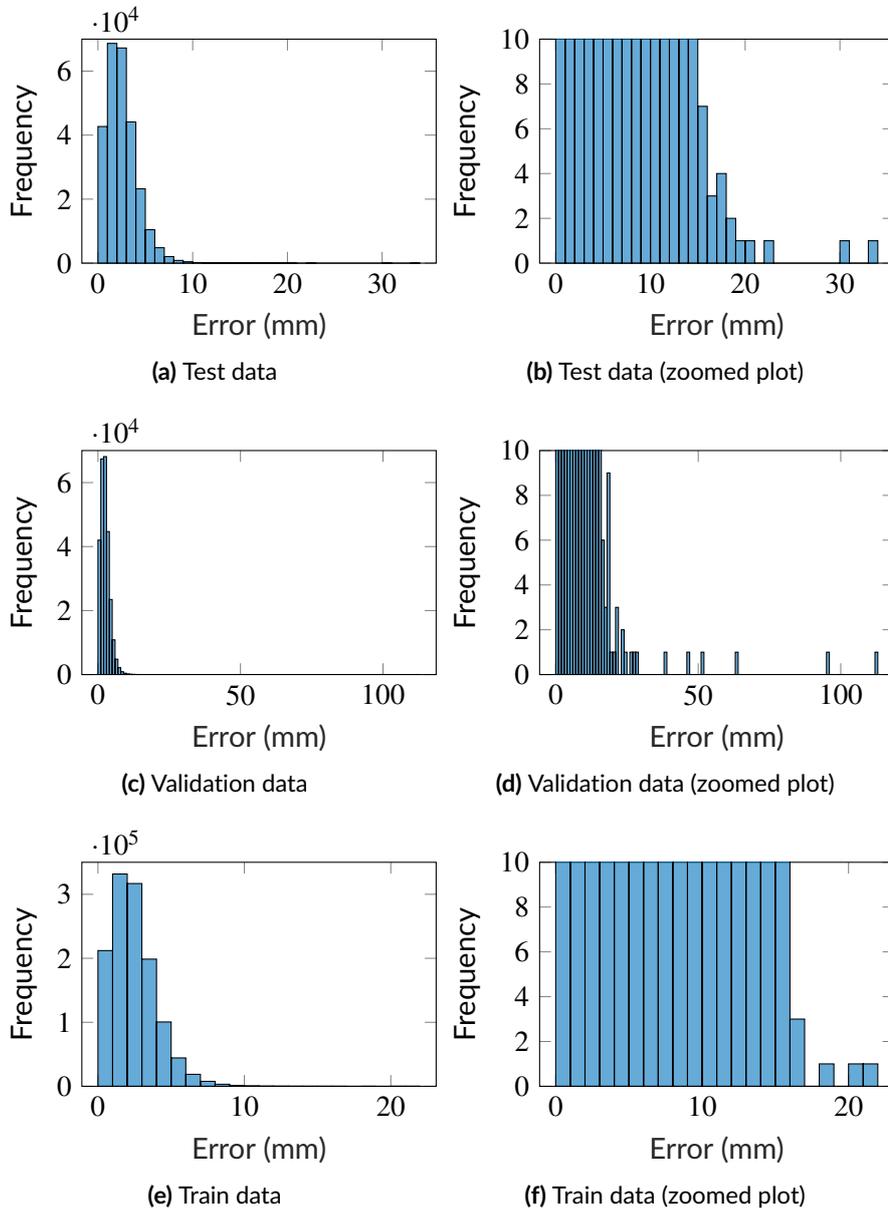


Figure A.15: Histogram of the localization error for the regression network. The histograms are given for each data partition.

A.8 Results on Clinical Data

Table A.3: Localization errors (geodesic distance between prediction and label) in mm for each sample in the clinical dataset. The suffix LV/RV in Patient ID depicts the respective ventricular half where the excitation was triggered.

Patient ID	002_RV	003_RV	004_LV	004_RV	005_LV	005_RV	006_LV	006_RV
FuzzyNet303	6	49	78	22	21	21	81	19
FuzzyNet93	5	40	71	15	26	23	95	48
RegressionNet	9	64	57	16	28	8	81	40
Patient ID	007_LV	007_RV	008_LV	008_RV	009_LV	009_RV	010_RV	011_RV
FuzzyNet303	102	37	64	12	91	21	12	32
FuzzyNet93	110	23	41	29	85	28	6	24
RegressionNet	73	28	8	23	54	40	10	28
Patient ID	012_LV	012_RV	013_RV	014_LV	014_RV	015_LV	015_RV	016_LV
FuzzyNet303	107	74	28	33	35	68	18	62
FuzzyNet93	49	66	29	45	36	95	9	78
RegressionNet	80	35	21	43	42	85	13	20
Patient ID	016_RV	017_LV	017_RV	018_LV	018_RV	019_LV	019_RV	020_LV
FuzzyNet303	14	83	55	55	28	57	10	69
FuzzyNet93	31	91	24	117	38	62	42	82
RegressionNet	17	45	24	20	47	21	42	102
Patient ID	020_RV	021_LV	021_RV	022_RV	023_LV	023_RV	024_LV	024_RV
FuzzyNet303	30	71	35	31	45	42	95	12
FuzzyNet93	19	46	54	12	96	39	81	41
RegressionNet	20	11	30	32	25	38	66	61
Patient ID	025_LV	025_RV	026_LV	026_RV	027_LV	027_RV	028_LV	028_RV
FuzzyNet303	81	82	101	104	86	26	104	27
FuzzyNet93	65	9	54	39	72	25	67	45
RegressionNet	27	13	73	57	50	17	22	40
Patient ID	029_LV	029_RV	030_LV	030_RV	031_LV	031_RV	032_LV	032_RV
FuzzyNet303	83	38	112	35	80	47	47	104
FuzzyNet93	46	38	125	38	80	12	53	22
RegressionNet	51	43	61	37	37	39	29	48
Patient ID	033_LV	033_RV	034_LV	034_RV	035_LV	035_RV	036_LV	036_RV
FuzzyNet303	79	7	76	30	32	13	50	5
FuzzyNet93	53	13	59	37	82	15	34	4
RegressionNet	34	53	41	51	52	19	55	10
Patient ID	037_LV	037_RV	038_RV					
FuzzyNet303	89	92	65					
FuzzyNet93	75	28	55					
RegressionNet	20	27	56					

References

- [1] E. Wilkins, L. Wilson, K. Wickramasinghe, et al., “European cardiovascular disease statistics 2017,” 1 2017.
- [2] S. Bansilal, J. M. Castellano, and V. Fuster, “Global burden of CVD: focus on secondary prevention of cardiovascular disease,” *International Journal of Cardiology*, vol. 201, pp. S1–S7, 1 2015.
- [3] L. Di Lullo, R. Rivera, V. Barbera, et al., “Sudden cardiac death and chronic kidney disease: From pathophysiology to treatment strategies,” *International Journal of Cardiology*, vol. 217, pp. 16–27, 8 2016.
- [4] R. S. Oliveira, S. Alonso, F. O. Campos, et al., “Ectopic beats arise from micro-reentries near infarct regions in simulations of a patient-specific heart model,” *Scientific reports*, vol. 8, no. 1, p. 16392, 11 2018.
- [5] S. Genovesi, M. G. Valsecchi, E. Rossi, et al., “Sudden death and associated factors in a historical cohort of chronic haemodialysis patients,” *Nephrology, Dialysis, Transplantation : Official Publication of the European Dialysis and Transplant Association - European Renal Association*, vol. 24, no. 8, pp. 2529–2536, 1 2009.
- [6] N. Joki, M. Tokumoto, N. Takahashi, et al., “Current perspectives on sudden cardiac death in hemodialysis patients,” *Contributions to Nephrology*, vol. 196, pp. 5–12, 1 2018.
- [7] P. A. Kalra, D. Green, and D. Poulidakos, “Arrhythmia in hemodialysis patients and its relation to sudden death,” *Kidney international*, vol. 93, no. 4, pp. 781–783, 4 2018.
- [8] D. Poulidakos, K. Hnatkova, S. Skampardon, et al., “Sudden cardiac death in dialysis: Arrhythmic mechanisms and the value of non-invasive electrophysiology,” *Frontiers in Physiology*, vol. 10, 1 2019.
- [9] M. C. G. Wong, J. M. Kalman, E. Pedagogos, et al., “Temporal distribution of arrhythmic events in chronic kidney disease: Highest incidence in the long interdialytic period,” *Heart Rhythm*, vol. 12, no. 10, pp. 2047–2055, 1 2015.
- [10] F. Sacher, L. Jesel, C. Borni-Duval, et al., “Cardiac rhythm disturbances in hemodialysis patients: Early detection using an implantable loop recorder and correlation with biological and dialysis parameters,” *JACC Clinical Electrophysiology*, vol. 4, no. 3, pp. 397–408, 3 2018.
- [11] A. Loewe, Y. Lutz, D. Nairn, et al., “Hypocalcemia-induced slowing of human sinus node pacemaking,” *Biophysical Journal*, 7 2019.
- [12] J. Khanagavi, T. Gupta, W. S. Aronow, et al., “Hyperkalemia among hospitalized patients and association between duration of hyperkalemia and outcomes,” *Archives of Medical Science AMS*, vol. 10, no. 2, pp. 251–7, 5 2014.
- [13] J. C. Adams, K. Srivathsan, and W. K. Shen, “Advances in management of premature ventricular contractions,” *Journal of Interventional Cardiac Electrophysiology : an International Journal of Arrhythmias and Pacing*, vol. 35, no. 2, pp. 137–149, 1 2012.

- [14] D. W. Zhu, J. D. Maloney, T. W. Simmons, et al., “Radiofrequency catheter ablation for management of symptomatic ventricular ectopic activity,” *The American College of Cardiology*, vol. 26, no. 4, pp. 843–849, 1 1995.
- [15] A. K. Yetisen, J. L. Martinez-Hurtado, B. Ünal, et al., “Wearables in medicine.” *Advanced materials (Deerfield Beach, Fla.)*, p. e1706910, 6 2018.
- [16] J. Tully, C. Dameff, and C. A. Longhurst, “Wave of wearables: Clinical management of patients and the future of connected medicine.” *Clinics in laboratory medicine*, vol. 40, no. 1, pp. 69–82, 3 2020.
- [17] R. F. Schmidt, F. Lang, and M. Heckmann, Eds., *Physiologie des Menschen*, Springer-Lehrbuch, vol. 31., überarb. und aktualisierte Aufl. Heidelberg: Springer, 1 2011.
- [18] ZooFari. Heart diagram-en.svg, 01/03/2014, 02:46. https://commons.wikimedia.org/wiki/File:Heart_diagram-en.svg. Licensed under the Creative Commons Attribution-Share Alike 3.0 Unported license.
- [19] Pedros.lol. File:ConductionsystemoftheheartwithoutHeart.svg, 19/06/2011, 06:21. <https://commons.wikimedia.org/wiki/File:ConductionsystemoftheheartwithoutHeart.svg>. Licensed under the Creative Commons Attribution-Share Alike 3.0 Unported license.
- [20] Jmarchn. File:Precordial Leads 2.svg, 08/02/2010, 08:01. https://commons.wikimedia.org/wiki/File:Precordial_Leads_2.svg. Licensed under the Creative Commons Attribution-Share Alike 3.0 Unported license.
- [21] E. Goldberger, “The aVL, aVR, and aVF leads: A simplification of standard lead electrocardiography,” *American Heart Journal*, vol. 24, no. 3, pp. 378–396, 1 1942.
- [22] G. D. Gargiulo, “True unipolar ECG machine for wilson central terminal measurements.” *BioMed research international*, vol. 2015, p. 586397, 1 2015.
- [23] H.-J. Trappe and H.-P. Schuster, “EKG-Kurs für Isabel,” 1 2017.
- [24] V. M. F. Meijborg, C. E. Conrath, T. Opthof, et al., “Electrocardiographic T wave and its relation with ventricular repolarization along major anatomical axes.” *Circulation. Arrhythmia and electrophysiology*, vol. 7, no. 3, pp. 524–31, 6 2014.
- [25] N. El-Sherif and G. Turitto, “Electrolyte disorders and arrhythmogenesis,” *Cardiology Journal*, vol. 18, no. 3, pp. 233–245, 1 2011.
- [26] K. Arastéh, *Innere medizin*, Duale Reihe. Stuttgart: Thieme, 1 2018.
- [27] YassineMrabet. File:Hemodialysis-en.svg, 17/01/2008, 17:03. <https://commons.wikimedia.org/wiki/File:Hemodialysis-en.svg>. Licensed under the Creative Commons Attribution-Share Alike 3.0 Unported license.
- [28] H.-Y. Ong, *Renal failure and replacement therapies*, Competency-Based Critical Care, S. Blakeley, Ed. London: Springer, 1 2008.
- [29] A. L. Hodgkin and A. F. Huxley, “A quantitative description of membrane current and its application to conduction and excitation in nerve,” *Journal of Physiology*, vol. 117, pp. 500–544, 1 1952.
- [30] A. Loewe, “Modeling human atrial patho-electrophysiology from ion channels to ECG : substrates, pharmacology, vulnerability, and P-waves,” PhD thesis, Karlsruhe, 1 2016.
- [31] K. H. W. J. ten Tusscher, D. Noble, P. J. Noble, et al., “A model for human ventricular tissue,” *Am. J. Physiol.*, vol. 286, pp. H1573–H1589, 1 2004.
- [32] K. H. W. J. ten Tusscher and A. V. Panfilov, “Alternans and spiral breakup in a human ventricular tissue model,” *American Journal of Physiology. Heart and Circulatory Physiology*, vol. 291, no. 3, pp. H1088–100, 1 2006.

- [33] Y. Himeno, K. Asakura, C. Y. Cha, et al., "A human ventricular myocyte model with a refined representation of excitation-contraction coupling," *Biophysical Journal*, vol. 109, no. 2, pp. 415–427, 1 2015.
- [34] J. Malmivuo and R. Plonsey, *Bioelectromagnetism: principles and applications of bioelectric and biomagnetic fields*. Oxford University Press, 1 1995.
- [35] K. Acsai, G. Antoons, L. Livshitz, et al., "Microdomain [ca(2)(+)] near ryanodine receptors as reported by 1-type ca(2)(+) and na+/ca(2)(+) exchange currents," *The Journal of Physiology*, vol. 589, no. Pt 10, pp. 2569–2583, 1 2011.
- [36] R. Hinch, J. L. Greenstein, A. J. Tanskanen, et al., "A simplified local control model of calcium-induced calcium release in cardiac ventricular myocytes," *Biophysical Journal*, vol. 87, no. 6, pp. 3723–36, 12 2004.
- [37] M. Stern, "Theory of excitation-contraction coupling in cardiac muscle," *Biophysical Journal*, vol. 63, pp. 497–517, 8 1992.
- [38] L. Tung, "A bidomain model for describing ischemic myocardial D-C potentials," PhD thesis, 1 1978.
- [39] M. Wallman, N. P. Smith, and B. Rodriguez, "A comparative study of graph-based, eikonal, and monodomain simulations for the estimation of cardiac activation times," *IEEE Transactions on Biomedical Engineering*, vol. 59, no. 6, pp. 1739–1748, 1 2012.
- [40] M. Sermesant, E. Konukoglu, H. Delingette, et al., "An anisotropic multi-front fast marching method for real-time simulation of cardiac electrophysiology," in *Functional Imaging and Modeling of the Heart*, F. Sachse and G. Seemann, Eds. Springer Berlin / Heidelberg, 1 2007, vol. 4466, pp. 160–169.
- [41] C. S. Henriquez, "Simulating the electrical behavior of cardiac tissue using the bidomain model," *Critical Reviews in Biomedical Engineering*, vol. 21, no. 1, pp. 1–77, 1 1993.
- [42] D. U. J. Keller, F. M. Weber, G. Seemann, et al., "Ranking the influence of tissue conductivities on forward-calculated ECGs," *IEEE Transactions on Biomedical Engineering*, vol. 57, no. 7, pp. 1568–1576, 1 2010.
- [43] O. Dössel, M. W. Krueger, F. M. Weber, et al., "Computational modeling of the human atrial anatomy and electrophysiology," *Medical & Biological Engineering & Computing*, vol. 50, no. 8, pp. 773–799, 1 2012.
- [44] A. Bolz and W. Urbaszek, *Technik in der Kardiologie: Eine interdisziplinäre Darstellung*. Berlin, Heidelberg: Springer, 1 2002.
- [45] C. Bishop, *Pattern recognition and machine learning*, Information science and statistics (Computer science). New York, NY: Springer, 1 2006.
- [46] A. J. Smola and B. Schölkopf, "A tutorial on support vector regression," *Statistics and Computing*, vol. 14, no. 3, pp. 199–222, 1 2004.
- [47] V. Vapnik, S. Golowich, and A. Smola, "Support vector method for function approximation, regression estimation and signal processing," *Advances in Neural Information Processing Systems*, vol. 9, pp. 281–287, 1 1996.
- [48] S. S. Haykin, *Neural networks and learning machines*, vol. 10. Upper Saddle River, NJ: Pearson, 2009.
- [49] G. Ian, B. Yoshua, and C. Aaron, *Deep learning*. MIT Press, 1 2016.
- [50] J. Nocedal and S. J. Wright, *Numerical optimization*, Springer series in operations research. New York: Springer, 1 2006.

- [51] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 1 2014.
- [52] D. MacKay, "A practical bayesian framework for backpropagation networks," *Neural computation*, vol. 4, no. 3, pp. 448–472, 1 1992.
- [53] Aphex34. File:Typical cnn.png, 17/12/2015, 03:04. https://commons.wikimedia.org/wiki/File:Typical_cnn.png. Licensed under the Creative Commons Attribution-Share Alike 4.0 International license.
- [54] E. Nilsson, A. Gasparini, J. Ärnlöv, et al., "Incidence and determinants of hyperkalemia and hypokalemia in a large healthcare system." *International Journal of Cardiology*, vol. 245, pp. 277–284, 10 2017.
- [55] D. Banerjee, "Sudden cardiac death in haemodialysis: clinical epidemiology and mechanisms." *Journal of Electrocardiology*, vol. 49, no. 6, pp. 843–847, 1 2016.
- [56] C. Corsi, M. Cortesi, G. Callisesi, et al., "Noninvasive quantification of blood potassium concentration from ECG in hemodialysis patients." *Scientific reports*, vol. 7, p. 42492, 2 2017.
- [57] Z. I. Attia, C. V. DeSimone, J. J. Dillon, et al., "Novel bloodless potassium determination using a signal-processed single-lead ECG." *Journal of the American Heart Association*, vol. 5, no. 1, 1 2016.
- [58] O. Z. Yasin, Z. Attia, J. J. Dillon, et al., "Noninvasive blood potassium measurement using signal-processed, single-lead ECG acquired from a handheld smartphone." *Journal of Electrocardiology*, vol. 50, no. 5, pp. 620–625, 1 2017.
- [59] C.-S. Lin, C. Lin, W.-H. Fang, et al., "A deep-learning algorithm (ECG12net) for detecting hypokalemia and hyperkalemia by electrocardiography: Algorithm development." *JMIR medical informatics*, vol. 8, no. 3, p. e15931, 3 2020.
- [60] N. Pilia, S. Severi, J. G. Raimann, et al., "Quantification and classification of potassium and calcium disorders with the electrocardiogram: What do clinical studies, modeling, and reconstruction tell us?" *APL Bioengineering*, vol. 4, no. 4, p. 041501, 12 2020.
- [61] A. Loewe, M. Hernandez Mesa, N. Pilia, et al., "A heterogeneous formulation of the himeno et al. human ventricular myocyte model for simulation of body surface ECGs," in *Computing in Cardiology*, vol. 45, 2018.
- [62] M. Hernandez Mesa, "Analysis of the effects of serum calcium changes on the ECG in a computational model," Bachelor's thesis, Karlsruhe, 1 2017.
- [63] M. Hernández Mesa, N. Pilia, O. Dössel, et al., "Effects of serum calcium changes on the cardiac action potential and the ECG in a computational model," in *Current Directions in Biomedical Engineering*, vol. 4, no. 1. De Gruyter, 1 2018, pp. 251–254.
- [64] T. O'Hara, L. Virag, A. Varro, et al., "Simulation of the undiseased human cardiac ventricular action potential: model formulation and experimental validation," *PLoS Computational Biology*, vol. 7, no. 5, p. e1002061, 1 2011.
- [65] S. Severi, C. Corsi, and E. Cerbai, "From in vivo plasma composition to in vitro cardiac electrophysiology and in silico virtual heart: the extracellular calcium enigma," *Philosophical Transactions. Series A, Mathematical, Physical, and Engineering Sciences*, vol. 367, no. 1896, pp. 2203–2223, 1 2009.
- [66] E. Passini and S. Severi, "Computational analysis of extracellular calcium effects on an improved human ventricular action potential model," in *2012 Computing in Cardiology*, 2012, pp. 873–876.

- [67] S. Severi, E. Grandi, C. Pes, et al., "Calcium and potassium changes during haemodialysis alter ventricular repolarization duration: in vivo and in silico analysis," *Nephrology, Dialysis, Transplantation : Official Publication of the European Dialysis and Transplant Association - European Renal Association*, vol. 23, no. 4, pp. 1378–1386, 1 2008.
- [68] C. Bartolucci, E. Passini, J. Hyttinen, et al., "Simulation of the effects of extracellular calcium changes leads to a novel computational model of human ventricular action potential with a revised calcium handling," *Frontiers in Physiology*, vol. 11, 1 2020.
- [69] D. U. J. Keller, D. L. Weiss, O. Dossel, et al., "Influence of $i(k_s)$ heterogeneities on the genesis of the t-wave: a computational evaluation." *IEEE Transactions on Bio-Medical Engineering*, vol. 59, no. 2, pp. 311–22, 2 2012.
- [70] G. Seemann, M. Alvarez de Eulate, N. Konrad, et al., "Evaluating body surface ECG differences of simulated long-QT syndromes," in *Computing in Cardiology Conference (CinC), 2013*, Zaragoza, Spain, 9 2013.
- [71] V. Elharrar and B. Surawicz, "Cycle length effect on restitution of action potential duration in dog cardiac fibers," *The American Journal of Physiology*, vol. 244, no. 6, pp. H782–92, 1 1983.
- [72] G. Seemann, F. B. Sachse, M. Karl, et al., "Framework for modular, flexible and efficient solving the cardiac bidomain equation using PETSc," *Mathematics in Industry*, vol. 15, no. 2, pp. 363–369, 1 2010.
- [73] G. Lenis, T. Baas, and O. Dössel, "Automatic detection and classification of ectopic beats in the ECG using a support vector machine," in *45. Jahrestagung der DGBMT im VDE. Proceedings BMT 2011*. Freiburg, Deutschland, 27-30 Sep. 2011: Jahrestagung der DGBMT - Deutschen Gesellschaft für Biomedizinische Technik im VDE, 1 2011.
- [74] N. Pilia, O. Dössel, G. Lenis, et al., "ECG as a tool to estimate potassium and calcium concentrations in the extracellular space," in *Computing in Cardiology*, vol. 44, 2017.
- [75] N. Pilia, "Characterization and reconstruction of ionic concentrations in the human ventricles analyzing the action potential and the surface ECG," Master's thesis, Karlsruhe, 2016.
- [76] I. B. Lee, S. C. Shin, Y. W. Jang, et al., "Comparison of conductive fabric sensor and ag-agCl sensor under motion artifacts." in *Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Annual International Conference*, vol. 2008, 1 2008, pp. 1300–3.
- [77] R. B. Chukka and C. S. Kumar, *Analysis and comparison of ECG signal quality assessments methods*, Advances in Intelligent Systems and Computing, 1 2020, vol. 989, pp. 875–881.
- [78] G. B. Moody, W. K. Muldrow, and R. G. Mark, "A noise stress test for arrhythmia detectors," in *Computers in Cardiology 1984*, vol. 11, 1984, pp. 381–384.
- [79] M. Hernández Mesa, N. Pilia, O. Dössel, et al., "Influence of ECG lead reduction techniques for extracellular potassium and calcium concentration estimation," in *Current Directions in Biomedical Engineering*, vol. 5, no. 1. De Gruyter, 1 2019, pp. 69–72.
- [80] N. Pilia, C. Ritter, D. Potyagaylo, et al., "Determination of the excitation origin in the ventricles from the ECG using support vector regression," in *Current Directions in Biomedical Engineering*, vol. 3, no. 2. de Gruyter, 1 2017, pp. 257–260.
- [81] N. Pilia, M. Hernandez Mesa, O. Dössel, et al., "ECG-based estimation of potassium and calcium concentrations: Proof of concept with simulated data," in *41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2019, pp. 2610–2613.

- [82] D. R. Hardoon, S. Szedmak, and J. Shawe-Taylor, "Canonical correlation analysis: an overview with application to learning methods." *Neural Computation*, vol. 16, no. 12, pp. 2639–64, 12 2004.
- [83] A. Mincholé, E. Zacur, R. Ariga, et al., "MRI-based computational torso/biventricular multi-scale models to investigate the impact of anatomical variability on the ECG QRS complex," *Frontiers in Physiology*, vol. 10, 1 2019.
- [84] M. Meinzer, "Patient-dependency of ionic concentration reconstruction and possible solutions," Master's thesis, Karlsruhe, 11 2020.
- [85] W. Kahlmann, "Development of a fast method to tailor purkinje activation based on QRS information," Master's thesis, Karlsruhe, 1 2017.
- [86] L. Cardone-Noott, A. Bueno-Orovio, A. Mincholé, et al., "Human ventricular activation sequence and the simulation of the electrocardiographic QRS complex and its variability in healthy and intraventricular block conditions." *Europace : European pacing, arrhythmias, and cardiac electrophysiology : journal of the working groups on cardiac pacing, arrhythmias, and cardiac cellular electrophysiology of the European Society of Cardiology*, vol. 18, no. suppl 4, pp. iv4–iv15, 12 2016.
- [87] J. D. Bayer, R. C. Blake, G. Plank, et al., "A novel rule-based algorithm for assigning myocardial fiber orientation to computational heart models," *Annals of Biomedical Engineering*, vol. 40, no. 10, pp. 2243–2254, 1 2012.
- [88] S. Schuler, J. D. Tate, T. F. Oostendorp, et al., "Spatial downsampling of surface sources in the forward problem of electrocardiography," in *Functional Imaging and Modeling of the Heart*, Lecture Notes in Computer Science, Y. Coudière, V. Ozenne, E. Vigmond, et al., Eds., vol. 11504. Springer International Publishing, 5 2019, pp. 29–36.
- [89] G. Moik geb. Schwaderlapp, "Reconstruction of the excitation organ in the ventricles using body surface potential maps," Master's thesis, Karlsruhe, 2019.
- [90] W. Bai, W. Shi, A. de Marvao, et al., "A bi-ventricular cardiac atlas built from 1000+ high resolution MR images of healthy subjects and an analysis of shape and motion," *Medical Image Analysis*, vol. 26, no. 1, pp. 133–145, 1 2015.
- [91] J. Bayer, A. J. Prassl, A. Pashaei, et al., "Universal ventricular coordinates: A generic framework for describing position within the heart and transferring data." *Medical Image Analysis*, vol. 45, pp. 83–93, 4 2018.
- [92] A. L. Goldberger, L. A. Amaral, L. Glass, et al., "Physiobank, physiokit, and physionet: components of a new research resource for complex physiologic signals." *Circulation*, vol. 101, no. 23, pp. E215–20, 6 2000.
- [93] R. Bousseljot, D. Kreiseler, and A. Schnabel, "Nutzung der EKG-Signaldatenbank CARDIODAT der PTB über das Internet," *Biomedizinische Technik/Biomedical Engineering*, pp. 317–318, 1 2009.
- [94] F. Odille, S. Liu, P. van Dam, et al., "Statistical variations of heart orientation in healthy adults," in *2017 Computing in Cardiology Conference (CinC)*, 2017 Computing in Cardiology Conference (CinC), vol. 44, 1 2017.
- [95] A. Neic, F. O. Campos, A. J. Prassl, et al., "Efficient computation of electrograms and ECGs in human whole heart simulations using a reaction-eikonal model," *Journal of Computational Physics*, vol. 346, pp. 191–211, 1 2017.
- [96] A. A. Danilov and A. S. Yurova, "Finite element method for forward ECG calculation," *Computational Mathematics and Mathematical Physics*, vol. 59, pp. 2033–2040, 2 2020.

- [97] D. Durrer, R. T. van Dam, G. E. Freud, et al., "Total excitation of the isolated human heart," *Circulation*, vol. 41, no. 6, pp. 899–912, 1 1970.
- [98] P. C. Hansen and P. C. O'Leary, "The use of the l-curve in the regularization of discrete ill-posed problems," *SIAM J. Scientific Computing*, vol. 14, no. 6, pp. 1487–1503, 1 1993.
- [99] N. V. Chawla, K. W. Bowyer, L. O. Hall, et al., "SMOTE: Synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, 1 2002.
- [100] B. Smardanski, "Diagnosing hypo- and hyperkalemia and hypo- and hypercalcemia with the 12-lead ECG," Master's thesis, Karlsruhe, 2019.
- [101] R. Astan, I. Akpınar, A. Karan, et al., "The effect of hemodialysis on electrocardiographic parameters." *Annals of Noninvasive Electrocardiology: the Official Journal of the International Society for Holter and Noninvasive Electrocardiology, Inc.*, vol. 20, no. 3, pp. 253–7, 5 2015.
- [102] O. Kinoshita, G. Kimura, S. Kamakura, et al., "Effects of hemodialysis on body surface maps in patients with chronic renal failure." *Nephron*, vol. 64, no. 4, pp. 580–6, 1 1993.
- [103] S. Becker, "Separating ECG changes caused by potassium channel blocks from changes by abnormal potassium concentrations," Bachelor's thesis, Karlsruhe, 2020.
- [104] P. A. Friedman, C. G. Scott, K. Bailey, et al., "Errors of classification with potassium blood testing: The variability and repeatability of critical clinical tests." *Mayo Clinic proceedings*, vol. 93, no. 5, pp. 566–572, 5 2018.
- [105] M. Wu, J. Chiang, Y. Yang, et al., "Predicting hyperkalemia by a two-staged artificial neural network," in *Computers in Cardiology, 2003*, 1 2003, pp. 433–435.
- [106] W. Tzeng, Y. Chan, and J. Hsieh, "Predicting hyperkalemia by the use of a 12-lead temporal-spatial electrocardiograph: clinical evaluations and model simulations," in *Computers in Cardiology, 2005*, 1 2005, pp. 215–218.
- [107] C. D. Galloway, A. V. Valys, J. B. Shreibati, et al., "Development and validation of a deep-learning model to screen for hyperkalemia from the electrocardiogram." *JAMA cardiology*, vol. 4, no. 5, pp. 428–436, 5 2019.
- [108] O. R. Segal, A. W. C. Chow, T. Wong, et al., "A novel algorithm for determining endocardial VT exit site from 12-lead surface ECG characteristics in human, infarct-related ventricular tachycardia." *Journal of Cardiovascular Electrophysiology*, vol. 18, no. 2, pp. 161–8, 2 2007.
- [109] R. D. Anderson, S. Kumar, R. Parameswaran, et al., "Differentiating right- and left-sided outflow tract ventricular arrhythmias." *Circulation. Arrhythmia and Electrophysiology*, vol. 12, no. 6, p. e007392, 6 2019.
- [110] O. Punshchykova, J. Švehlíková, M. Tyšler, et al., "Influence of torso model complexity on the noninvasive localization of ectopic ventricular activity," *Measurement Science Review*, vol. 16, no. 2, pp. 96–102, 1 2016.
- [111] H. Kaiyue, N. Zhenning, Z. Gaoyan, et al., "Localization of origins of premature ventricular contraction in the whole ventricle based on machine learning and automatic beat recognition from 12-lead ECG," *Physiological Measurement*, 1 2020.
- [112] T. Yang, L. Yu, Q. Jin, et al., "Localization of origins of premature ventricular contraction by means of convolutional neural network from 12-lead ECG." *IEEE Transactions on Biomedical Engineering*, vol. 65, no. 7, pp. 1662–1671, 7 2018.
- [113] T. Yang and B. He, "System and method for localization of origins of cardiac arrhythmia using electrocardiography and neural networks," US Patent, 3 2019.

- [114] D. Potyagaylo, W. H. W. Schulze, and O. Dössel, “A new method for choosing the regularization parameter in the transmembrane potential based inverse problem of ECG,” in *Computing in Cardiology*, 9-12 Sept. Krakow, Poland, 1 2012, pp. 29–32.
- [115] D. Potyagaylo, E. G. Cortés, W. H. W. Schulze, et al., “Binary optimization for source localization in the inverse problem of ECG,” *Medical & Biological Engineering & Computing*, vol. 52, no. 9, pp. 717–728, 1 2014.
- [116] W. H. W. Schulze, “ECG imaging of ventricular activity in clinical applications,” PhD thesis, Karlsruhe, 1 2015.
- [117] A. de Marvao, T. J. W. Dawes, W. Shi, et al., “Population-based studies of myocardial hypertrophy: high resolution cardiovascular magnetic resonance atlases improve statistical power.” *Journal of cardiovascular magnetic resonance : official journal of the Society for Cardiovascular Magnetic Resonance*, vol. 16, p. 16, 2 2014.
- [118] baiwenjia. Statistical-shape-model. <https://github.com/UK-Digital-Heart-Project/Statistical-Shape-Model>. Statistical shape model for the ventricles of the heart.
- [119] S. Schuler. Fim eikonal. https://github.com/KIT-IBT/FIM_Eikonal. Extension of SCI-Solver Eikonal for simulation of anisotropic wavefront propagation in the heart.
- [120] A. L. Goldberger, L. A. Amaral, L. Glass, et al., “Physiobank, physiokit, and physionet: components of a new research resource for complex physiologic signals,” *Circulation*, vol. 101, no. 23, pp. E215–20, 1 2000.
- [121] A. Paszke, S. Gross, F. Massa, et al., “Pytorch: An imperative style, high-performance deep learning library,” in *Advances in Neural Information Processing Systems 32*, W. H., L. H., B. A., et al., Eds. Curran Associates, Inc., 1 2019, pp. 8024–8035.
- [122] S. Xie, R. Girshick, P. Dollár, et al., “Aggregated residual transformations for deep neural networks,” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1492–1500, 11 2016.
- [123] L. N. Smith, “Cyclical learning rates for training neural networks,” *arXiv:1506.01186v6 [cs.CV]*, 6 2015.
- [124] D. Potyagaylo, M. Chmelevsky, S. Zubarev, et al., “Evaluation of ECGI localization accuracy for single pacings in CRT patients,” in *Computing in Cardiology*, vol. 45, 9 2018.
- [125] S. Schuler, D. Potyagaylo, and O. Dössel, “Using a spatio-temporal basis for ECG imaging of ventricular pacings: Insights from simulations and first application to clinical data,” in *41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 7 2019, pp. 1559–1562.
- [126] Y. Rudy, “The forward problem of electrocardiography revisited.” *Circulation. Arrhythmia and Electrophysiology*, vol. 8, no. 3, pp. 526–8, 6 2015.
- [127] G. Luongo, S. Schuler, A. Luik, et al., “Non-invasive characterization of atrial flutter mechanisms using recurrence quantification analysis on the ECG: a computational study.” *IEEE Transactions on Biomedical Engineering*, 5 2020.
- [128] K. He, X. Zhang, S. Ren, et al., “Deep residual learning for image recognition,” *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 12 2015.
- [129] T. Akiba, S. Sano, T. Yanase, et al., “Optuna: A next-generation hyperparameter optimization framework,” in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 1 2019, pp. 2623–2631.
- [130] S. Hochreiter and J. Schmidhuber, “Long short-term memory.” *Neural computation*, vol. 9, no. 8, pp. 1735–80, 11 1997.

- [131] M. Baker, “1,500 scientists lift the lid on reproducibility.” *Nature*, vol. 533, no. 7604, pp. 452–4, 5 2016.
- [132] M. McNutt, “Journals unite for reproducibility.” *Science (New York, N.Y.)*, vol. 346, no. 6210, p. 679, 11 2014.
- [133] D. C. Ince, L. Hatton, and J. Graham-Cumming, “The case for open computer programs.” *Nature*, vol. 482, no. 7386, pp. 485–8, 2 2012.
- [134] G. Lenis and O. Dössel, “T wave morphology during heart rate turbulence in patients with chronic heart failure,” in *Biomedizinische Technik. Biomedical Engineering*, vol. 58, no. s1, 1 2013.
- [135] G. Lenis, N. Pilia, T. Oesterlein, et al., “P wave detection and delineation in the ECG based on the phase free stationary wavelet transform and using intracardiac atrial electrograms as reference,” *Biomedizinische Technik. Biomedical Engineering*, vol. 61, no. 1, pp. 37–56, 1 2016.
- [136] G. Lenis, P. Reichensperger, D. Sommer, et al., “Detection of microsleap events in a car driving simulation study using electrocardiographic features,” in *Current Directions in Biomedical Engineering*, vol. 2, no. 1. De Gruyter, 1 2016, pp. 283–287.
- [137] G. Lenis, M. Kircher, J. Lazaro, et al., “Separating the effect of respiration on the heart rate variability using granger’s causality and linear filtering,” *Biomedical Signal Processing and Control*, vol. 31, pp. 272–278, 1 2017.
- [138] G. Lenis, N. Pilia, A. Loewe, et al., “Comparison of baseline wander removal techniques considering the preservation of ST changes in the ischemic ECG: A simulation study,” *Computational and Mathematical Methods in Medicine*, vol. 2017, p. 9295029, 1 2017.
- [139] A. Lyon, A. Mincholé, J. P. Martínez, et al., “Computational techniques for ECG analysis and interpretation in light of their contribution to medical advances.” *Journal of the Royal Society, Interface*, vol. 15, no. 138, 1 2018.
- [140] S. Majumder, L. Chen, O. Marinov, et al., “Noncontact wearable wireless ECG systems for long-term monitoring.” *IEEE Reviews in Biomedical Engineering*, vol. 11, pp. 306–321, 1 2018.
- [141] J. B. Nielsen, J. T. Kuhl, A. Pietersen, et al., “P-wave duration and the risk of atrial fibrillation: Results from the copenhagen ECG study,” *Heart Rhythm : the Official Journal of the Heart Rhythm Society*, vol. 12, no. 9, pp. 1887–1895, 1 2015.
- [142] A. Loewe, M. W. Krueger, F. Holmqvist, et al., “Influence of the earliest right atrial activation site and its proximity to interatrial connections on P-wave morphology,” *Europace*, vol. 18, no. suppl 4, pp. iv35–iv43, 1 2016.
- [143] R. Andlauer, G. Seemann, L. Baron, et al., “Influence of left atrial size on P-wave morphology: differential effects of dilation and hypertrophy,” *Europace*, vol. 20, no. S3, pp. iii36–iii44, 11 2018.
- [144] V. Jahmunah, S. L. Oh, J. K. E. Wei, et al., “Computer-aided diagnosis of congestive heart failure using ECG signals - a review.” *Physica medica : PM : an international journal devoted to the applications of physics to medicine and biology : official journal of the Italian Association of Biomedical Physics (AIFB)*, vol. 62, pp. 95–104, 6 2019.
- [145] G. Lenis, “Signal processing methods for the analysis of the electrocardiogram,” PhD thesis, 1 2017.
- [146] A. J. Demski and M. Llamedo Soria, “ecg-kit a matlab toolbox for cardiovascular signal processing,” *Journal of Open Research Software*, vol. 4, 1 2016.

- [147] P. Laguna, R. Jané, and P. Caminal, "Automatic detection of wave boundaries in multilead ECG signals: Validation with the CSE database," *Computers and Biomedical Research*, vol. 27, no. 1, pp. 45–60, 1 1994.
- [148] N. Pilia, C. Nagel, G. Lenis, et al., "ECGdeli - an open source ECG delineation toolbox for MATLAB," *SoftwareX*, vol. 13, p. 100639, 1 2021.
- [149] N. Pilia, C. Nagel, G. Lenis, et al. (2020) ECGdeli repository - an open source ECG delineation toolbox for MATLAB. Available on Zenodo. <https://doi.org/10.5281/zenodo.3944621>.
- [150] P. Laguna, R. G. Mark, A. Goldberg, et al., "A database for evaluation of algorithms for measurement of QT and other waveform intervals in the ECG," in *Computers in Cardiology 1997*, 1 1997, pp. 673–676.
- [151] L. Y. Di Marco and L. Chiari, "A wavelet-based ECG delineation algorithm for 32-bit integer online processing," *Biomedical Engineering Online*, vol. 10, p. 23, 1 2011.
- [152] N. Spicher and M. Kukuk, "Delineation of electrocardiograms using multiscale parameter estimation." *IEEE journal of biomedical and health informatics*, vol. 24, no. 8, pp. 2216–2229, 8 2020.
- [153] G. Lenis, T. Heine, P. Reichensperger, et al., "Electrocardiographic features for the measurement of drivers' mental workload," *Applied Ergonomics*, vol. 61, no. May, pp. 31–43, 1 2017.
- [154] L. Sörnmo and P. Laguna, *Bioelectrical Signal Processing in Cardiac and Neurological Applications*. Amsterdam: Elsevier, 1 2005.
- [155] U. R. Acharya, H. Fujita, O. S. Lih, et al., "Automated detection of arrhythmias using different intervals of tachycardia ECG segments with convolutional neural network," *Information Sciences*, vol. 405, pp. 81–90, 1 2017.
- [156] U. R. Acharya, H. Fujita, S. L. Oh, et al., "Application of deep convolutional neural network for automated detection of myocardial infarction using ECG signals," *Information Sciences*, vol. 415-416, pp. 190–198, 1 2017.
- [157] K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep inside convolutional networks: Visualising image classification models and saliency maps." *arXiv preprint arXiv:1312.6034*, 2013.
- [158] A. Loewe, W. H. W. Schulze, Y. Jiang, et al., "ECG-based detection of early myocardial ischemia in a computational model: Impact of additional electrodes, optimal placement, and a new feature for ST deviation," *BioMed Research International Article*, vol. 2015, p. 530352, 1 2014.
- [159] M. Lindner, "Machine learning approaches for a 20 AFI classification using BSPM and RQA images," Master's thesis, 2020.
- [160] S. Pollnow, N. Pilia, G. Schwaderlapp, et al., "An adaptive spatio-temporal gaussian filter for processing cardiac optical mapping data," *Computers in Biology and Medicine*, vol. 102, pp. 267–277, 6 2018.
- [161] T. Oesterlein, O. Dössel, D. Frisch, et al., "Method and system for determining ventricular far field contribution in atrial electrograms," Patent, 2019.
- [162] T. Zielinski, "Instantaneous phase shift estimation methods," in *IEEE Instrumentation and Measurement Technology Conference and IMEKO Technical Committee 7. Conference Proceedings*, Quality Measurement: The Indispensable Bridge between Theory and Reality (No Measurements? No Science! Joint Conference), vol. 1, 1 1996, pp. 162–167.

-
- [163] P. Kuklik, S. Zeemering, B. Maesen, et al., “Reconstruction of instantaneous phase of unipolar atrial contact electrogram using a concept of sinusoidal recombination and hilbert transform,” *IEEE Transactions on Biomedical Engineering*, vol. 62, no. 1, pp. 296–302, 1 2015.
- [164] S. Süß, “Determination of the information gain by estimating the instantaneous phase of biosignals,” Bachelor’s thesis, Karlsruhe, 2018.
- [165] C. Nagel, “Robust conduction velocity estimation for a clinical setting,” Master’s thesis, Karlsruhe, 2019.
- [166] C. Nagel, N. Pilia, L. Unger, et al., “Performance of different atrial conduction velocity estimation algorithms improves with knowledge about the depolarization pattern,” in *Current Directions in Biomedical Engineering*, vol. 5, no. 1. De Gruyter, 1 2019, pp. 101–104.
- [167] J. Koch, “Enhancing conduction velocity estimation by atrial electrogram analysis,” Bachelor’s thesis, Karlsruhe, 2020.
- [168] G. Schwaderlapp, “Definition, estimation, validation and clinical value of the dominant frequency of intracardiac atrial electrograms,” Bachelor’s thesis, Karlsruhe, 1 2017.
- [169] N. Pilia, G. Schwaderlapp, T. Oesterlein, et al., “Evaluating regularity of intracardiac electrograms for determining the health condition of a patient,” in *Workshop Biosignale 2018 Erfurt*, 2018.
- [170] A. Loewe, E. Poremba, T. G. Oesterlein, et al., “An interactive virtual reality environment for analysis of clinical atrial arrhythmias and ablation planning,” in *2017 Computing in Cardiology Conference (CinC)*, 2017 Computing in Cardiology Conference (CinC), vol. 44, 1 2017.

List of Publications and Supervised Theses

Journal Articles

First authorship and shared first authorship

- **N. Pilia**, C. Nagel, G. Lenis, S. Becker, O. Dössel, and A. Loewe, “ECGdeli - An Open Source ECG Delineation Toolbox for MATLAB,” *Software X*, 2020, accepted, in press.
- **N. Pilia**, S. Severi, J. G. Raimann, S. Genovesi, O. Dössel, P. Kotanko, C. Corsi, and A. Loewe, “Quantification and classification of potassium and calcium disorders with the electrocardiogram: What do clinical studies, modeling, and reconstruction tell us?,” *APL Bioengineering*, vol. 4, no. 4, p. 041501, 2020.
- S. Pollnow, **N. Pilia**, G. Schwaderlapp, A. Loewe, O. Dössel, and G. Lenis, “An adaptive spatio-temporal Gaussian filter for processing cardiac optical mapping data,” *Comput Biol Med*, vol. 102, pp. 267–277, 2018.
- G. Lenis, **N. Pilia**, A. Loewe, W. H. W. Schulze, and O. Dössel, “Comparison of Baseline Wander Removal Techniques considering the Preservation of ST Changes in the Ischemic ECG: A Simulation Study,” *Computational and Mathematical Methods in Medicine*, vol. 2017, p. 9295029, 2017.
- G. Lenis, **N. Pilia**, T. Oesterlein, A. Luik, C. Schmitt, and O. Dössel, “P wave detection and delineation in the ECG based on the phase free stationary wavelet transform and using intracardiac atrial electrograms as reference,” *Biomed Tech (Berl)*, vol. 61, no. 1, pp. 37–56, 2016.
- S. Schuler, **N. Pilia**, G. Moik, M. Rees, A. Loewe, and O. Dössel, “Deep Learning Does Not Need Patient-Specific Geometries to Localize Ventricular Excitation Origins from ECGs,” writing in progress.
- S. Schuler, **N. Pilia**, G. Moik, O. Dössel, and A. Loewe, “Cobiveco: Consistent biventricular coordinates for precise and intuitive description of local position in the heart – with MATLAB implementation,” *Medical Image Analysis*, under review.

Refereed Conference Articles

- C. Nagel, **N. Pilia**, A. Loewe, and O. Dössel, “Quantification of Interpatient 12-lead ECG Variabilities within a Healthy Cohort,” in *Current Directions in Biomedical Engineering*, 2020, vol. 6, no. 3, pp. 493–496.
- **N. Pilia**, M. Hernandez Mesa, O. Dössel, and A. Loewe, “ECG-based Estimation of Potassium and Calcium Concentrations: Proof of Concept with Simulated Data,” in *41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2019, pp. 2610–2613.
- C. Nagel, **N. Pilia**, L. Unger, and O. Dössel, “Performance of Different Atrial Conduction Velocity Estimation Algorithms Improves with Knowledge about the Depolarization Pattern,” in *Current Directions in Biomedical Engineering*, 2019, vol. 5, no. 1, pp. 101–104.
- M. Hernández Mesa, **N. Pilia**, O. Dössel, and A. Loewe, “Influence of ECG Lead Reduction Techniques for Extracellular Potassium and Calcium Concentration Estimation,” in *Current Directions in Biomedical Engineering*, 2019, vol. 5, no. 1, pp. 69–72.
- A. Loewe, M. Hernandez Mesa, **N. Pilia**, S. Severi, and O. Dössel, “A Heterogeneous Formulation of the Himeno et al. Human Ventricular Myocyte Model for Simulation of Body Surface ECGs,” in *Computing in Cardiology*, 2018, vol. 45.
- M. Hernández Mesa, **N. Pilia**, O. Dössel, S. Severi, and A. Loewe, “Effects of Serum Calcium Changes on the Cardiac Action Potential and the ECG in a Computational Model,” in *Current Directions in Biomedical Engineering*, 2018, vol. 4, no. 1, pp. 251–254.
- **N. Pilia**, C. Ritter, D. Potyagaylo, W. H. W. Schulze, O. Dössel, and G. Lenis, “Determination of the excitation origin in the ventricles from the ECG using support vector regression,” in *Current Directions in Biomedical Engineering*, 2017, vol. 3, no. 2, pp. 257–260.
- **N. Pilia**, O. Dössel, G. Lenis, and A. Loewe, “ECG as a Tool to Estimate Potassium and Calcium Concentrations in the Extracellular Space,” in *Computing in Cardiology*, 2017, vol. 44.
- A. Loewe, E. Poremba, T. G. Oesterlein, **N. Pilia**, M. Pfeiffer, O. Doessel, and S. Speidel, “An Interactive Virtual Reality Environment for Analysis of Clinical Atrial Arrhythmias and Ablation Planning,” in *2017 Computing in Cardiology Conference (CinC)*, vol. 44, 2017.
- **N. Pilia**, G. Lenis, A. Loewe, W. H. W. Schulze, and O. Dössel, “The impact of baseline wander removal techniques on the ST segment in simulated ischemic 12-lead ECG,” in *Current Directions in Biomedical Engineering*, 2015, vol. 1, no. 1, pp. 96–99.

Refereed Conference Abstracts

- **N. Pilia**, C. Corsi, S. Severi, O. Dössel, and A. Loewe, “Reconstruction of Potassium Concentrations with the ECG on Imbalanced Datasets,” in Workshop Biosignale 2020 Kiel, Germany, 2020.
- **N. Pilia**, O. Dössel, “Reconstruction of the Excitation Origin Region of Focal Ventricular Tachycardia with the ECG,” BMT 2018 - 52nd Annual Conference of the German Society for Biomedical Engineering, Aachen, Germany, 2018.
- **N. Pilia**, G. Schwaderlapp, T. Oesterlein, G. Lenis, A. Luik, C. Schmitt, and O. Dössel, “Evaluating Regularity of Intracardiac Electrograms for Determining the Health Condition of a Patient,” in Workshop Biosignale 2018 Erfurt, Germany, 2018.
- **N. Pilia**, G. Lenis, and O. Dössel, “Developing a robust method to delineate the P wave using information from intracardiac electrograms,” in Biosignalverarbeitung und Magnetische Methoden in der Medizin. Proceedings BBS 2014, Berlin, Germany, 2014.

Patents

- T. Oesterlein, O. Dössel, D. Frisch, A. Loewe, G. Lenis, and **N. Pilia**: Method and system for determining ventricular far field contribution in atrial electrograms,” 2019.

Reports and Theses

- **N. Pilia**, “Characterization and reconstruction of ionic concentrations in the human ventricles analyzing the action potential and the surface ECG”, Institute of Biomedical Engineering, Karlsruhe Institute of Technology (KIT), Master’s Thesis, 2016.
- **N. Pilia**, “A robust method to detect and characterise the P wave in the electrocardiogram”, Institute of Biomedical Engineering, Karlsruhe Institute of Technology (KIT), Bachelor’s Thesis, 2013.

Supervised Student Theses

- Maike Rees: Explainable Machine Learning and Semantic Features for the Localization of Focal Sources in the Ventricles, Master’s Thesis, Institute of Biomedical Engineering, Karlsruhe Institute of Technology (KIT), 2021.

- Michael Meinzer: Patient-dependency of ionic concentration reconstruction and possible solutions, Master's Thesis, Institute of Biomedical Engineering, Karlsruhe Institute of Technology (KIT), 2020.
- Moritz Linder: Machine learning approaches for a 20 AFI classification using BSPM and RQA images, Master's Thesis, Institute of Biomedical Engineering, Karlsruhe Institute of Technology (KIT), 2020 (Co-Supervision).
- Silvia Becker: Separating ECG changes caused by potassium channel blocks from changes by abnormal potassium concentrations, Bachelor's Thesis, Institute of Biomedical Engineering, Karlsruhe Institute of Technology (KIT), 2020.
- Jeanne Koch: Enhancing Conduction Velocity Estimation by Atrial Electrogram Analysis, Bachelor's Thesis, Institute of Biomedical Engineering, Karlsruhe Institute of Technology (KIT), 2020.
- Gerald Moik: Reconstruction of the Excitation Origin in the Ventricles using Body Surface Potential Maps, Master's Thesis, Institute of Biomedical Engineering, Karlsruhe Institute of Technology (KIT), 2020.
- Claudia Nagel: Robust conduction velocity estimation in a clinical setting, Master's Thesis, Institute of Biomedical Engineering, Karlsruhe Institute of Technology (KIT), 2019.
- Bilian Smardanski: Diagnosing hypo- and hyperkalaemia and hypo- and hypercalcaemia with the 12-lead ECG, Master's Thesis, Institute of Biomedical Engineering, Karlsruhe Institute of Technology (KIT), 2018.
- María Hernández Mesa: Analysis of the Effects of Serum Calcium Changes on the ECG in a Computational Model, Bachelor's Thesis, Institute of Biomedical Engineering, Karlsruhe Institute of Technology (KIT), 2018.
- Pedro Alvarez: Classification of atrial ectopic origins into spatial segments based on the 12-lead ECG, Master's Thesis, Institute of Biomedical Engineering, Karlsruhe Institute of Technology (KIT), 2018 (Co-Supervision).
- Simon Süß: Determination of the information gain by estimating the instantaneous phase of biosignals, Bachelor's Thesis, Institute of Biomedical Engineering, Karlsruhe Institute of Technology (KIT), 2017.

Supervised Student Research Projects

- María Hernández Mesa: Influence of Ionic Concentrations on the ECG Evaluated with a Computer Model of the Heart, Student Research Project, Institute of Biomedical Engineering, Karlsruhe Institute of Technology (KIT), 2018-2019.
- Rittwik Sood: Evaluation of methods for phase detection, Student Research Project, Institute of Biomedical Engineering, Karlsruhe Institute of Technology (KIT), 2017.
- Rafi Beinhorn: Erstellung und Auswertung eines Datensatzes mit Intrakardialen Signalen zur Auswertung von Komplexitätsmaßen, Student Research Project, Institute of Biomedical Engineering, Karlsruhe Institute of Technology (KIT), 2017.

Awards & Grants

- Third prize Student Presentation Competition, Workshop “Innovative Verarbeitung bioelektrischer und biomagnetischer Signale” BBS 2014, Berlin, Germany. **N. Pilia**, G. Lenis, O. Dössel: Developing a new method to delineate the P wave using information from intracardiac electrograms.
- Best Poster Award, Workshop Biosignalverarbeitung 2020, Kiel, Germany. **N. Pilia**, C. Corsi, S. Severi, O. Dössel, and A. Loewe: Reconstruction of Potassium Concentrations with the ECG on Imbalanced Datasets.
- Nvidia Higher Education and Research Grants, GPU Grant, 2019.
- Best paper award, DGBMT annual conference 2019, Frankfurt, Germany. C. Nagel, **N. Pilia**, L. Unger, O. Dössel: Performance of Different Atrial Conduction Velocity Estimation Algorithms Improves with Knowledge about the Depolarization Pattern.
- Rosanna Degani Young Investigator Award Semi-finalist, Computing in Cardiology 2017, Rennes, France. **N. Pilia**, O. Dössel, G. Lenis, A. Loewe, ECG as a Tool to Estimate Potassium and Calcium Concentrations in the Extracellular Space.
- Bill and Gary Sanders Poster Award, Conference Computing in Cardiology 2017, Rennes, France. A. Loewe, E. Poremba, T. G. Oesterlein, **N. Pilia**, Micha Pfeiffer, O. Dössel, S. Speidel: An Interactive Virtual Reality Environment for Analysis of Clinical Atrial Arrhythmias and Ablation Planning.

