Sensor Applications for Human Activity Recognition in Smart Environments

Vom Fachbereich Informatik
der Technischen Universität Darmstadt
genehmigte

DISSERTATION

zur Erlangung des akademischen Grades eines
Doktor-Ingenieurs (Dr.-Ing.)
von

Biying Fu

Erstgutachter:	Prof. Dr. Arjan Kuijper Technische Universität Darmstadt
Zweitgutachter:	Prof. Dr. techn. Dr.-Ing. eh. Dieter W. Fellner Technische Universität Darmstadt
Drittgutachter:	Prof. Dr. Kristof Van Laerhoven Universität Siegen
Tag der Einreichung:	24/09/2020
Tag der mündlichen Prüfung:	17/11/2020

Darmstädter Dissertation
D 17

Fu, Biying : Sensor Applications for Human Activity Recognition in Smart Environments
Darmstadt, Technische Universität Darmstadt,
Jahr der Veröffentlichung der Dissertation auf TUprints: 2021
URN: urn:nbn:de:tuda-tuprints-174858
Tag der mündlichen Prüfung: 17.11.2020

Erklärung zur Dissertation

Hiermit versichere ich die vorliegende Dissertation selbständig nur mit den angegebenen Quellen und Hilfsmitteln angefertigt zu haben. Alle Stellen, die aus Quellen entnommen wurden, sind als solche kenntlich gemacht. Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Darmstadt, den 24.09.2020 Biying Fu

Abstract

Human activity recognition (HAR) is the automated recognition of individual or group activities from sensor inputs. It deals with a wide range of application areas, such as for health care, assisting technologies, quantified-self and safety applications. HAR is the key to build human-centred applications and enables users to seamlessly and naturally interact with each other or with a smart environment. A smart environment is an instrumented room or space equipped with sensors and actuators to perceive the physical state or human activities within this space. The diversity of sensors makes it difficult to use the appropriate sensor to build specific applications. This work aims at presenting sensor-driven applications for human activity recognition in smart environments by using novel sensing categories beyond the existing sensor technologies commonly applied to these tasks. The intention is to improve the interaction for various sub-fields of human activities. Each application addresses the difficulties following the typical process pipeline for designing a smart environment application.

At first, I survey most prominent research works with focus on sensor-driven categorization in the research domain of HAR to identify possible research gaps to position my work. I identify two use-cases: quantified-self and smart home applications. Quantified-self aims at self-tracking and self-knowledge through numbers. Common sensor technology for daily tracking of various aerobic endurance training activities, such as *walking, running*, or *cycling* are based on acceleration data with wearable. However, more stationary exercises, such as strength-based training or stretching are also important for a healthy life-style, as they improve body coordination and balance. These exercises are not well tracked by wearing only a single wearable sensor, as these activities rely on coordinated movement of the entire body. I leverage two sensing categories to design two portable mobile applications for remote sensing of these more stationary exercises of physical workout.

Sensor-driven applications for smart home domain aim at building systems to make the life of the occupants safer and more convenient. In this thesis, I target at stationary applications to be integrated into the environment to allow a more natural interaction between the occupant and the smart environment. I propose two possible solutions to achieve this task. The first system is a surface acoustic based system which provides a sparse sensor setup to detect a basic set of activities of daily living including the investigation of minimalist sensor arrangement. The second application is a tag-free indoor positioning system. Indoor localization aims at providing location information to build intelligent services for smart homes. Accurate indoor position offers the basic context for high-level reasoning system to achieve more complex contexts. The floor-based localization system using electrostatic sensors is scalable to different room geometries due to its layout and modular composition. Finally, privacy with non-visual input is the main aspect for applications proposed in this thesis.

In addition, this thesis addresses the issue of adaptivity from prototypes towards real-world applications. I identify the issues of data sparsity in the training data and data diversity in the real-world data. In order to solve the issue of data sparsity, I demonstrate the data augmentation strategy to be applied on time series to increase the amount of training data by generating synthetic data. Towards mitigating the inherent difference of the development dataset and the real-world scenarios, I further investigate several approaches including metric-based learning and fine-tuning. I explore these methods to finetune the trained model on limited amount of individual data with and without retrain the pre-trained inference model. Finally some examples are stated as how to deploy the offline model to online processing device with limited hardware resources.

Zusammenfassung

Bei der automatischen Erkennung menschlicher Aktivitäten geht es darum den Zustand und die Bewegungen von einzelnen Personen oder auch Gruppen mit Hilfen von Sensoren zu detektieren. Für die Erkennung menschlicher Aktivitäten gibt es sehr breite Anwendungsbereiche, wie zum Beispiel im Gesundheitswesen, sämtliche assistierende Anwendungen, Quantified-Self oder auch sicherheitsrelevante Anwendungen. Sie ist ein Schlüssel für den Entwurf von auf den Menschen bezogene Anwendungen. Sie erlaubt eine einfache und natürliche Interaktion zwischen Nutzern untereinander, aber auch zwischen dem Nutzer und dessen intelligenter Umgebung. Unter einer intelligenten Umgebung versteht man dabei einen Raum, der mit untereinander vernetzten Sensoren und Aktoren ausgestattet ist. Diese kann den physikalischen Zustand und die menschlichen Aktivitäten innerhalb dieses Bereiches erkennen und analysieren. Die große Diversität der Sensoren macht es jedoch schwierig den geeigneten Sensortyp für einen bestimmten Anwendungsbereich auszuwählen. Sensorspezifische Limitierungen können ein ausschlaggebendes Auswahlkriterium sein. In dieser Arbeit beschäftige ich mich mit der sensor-getriebenen Entwicklung von Applikationen für die automatische Erkennung menschlicher Aktivitäten. Diese sollen eine verbesserte Interaktion mit der intelligenten Umgebung ermöglichen. Der Entwurf und die Ausgestaltung der jeweiligen Anwendung folgen der von mir vorgestellten Prozesskette einer Produktentwicklung. Basierend auf diese Prozesskette resultieren die wissenchenschaftliche Fragestellungen, die in dieser Arbeit beantwortet werden.

Ich beschäftige mich zunächst mit einer umfangreichen Untersuchung vorangegangener Arbeiten zur Kategorisierung von Sensoren. Die auf diesen Sensorkategorien basierenden Systeme zur Erkennung menschlicher Aktivitäten stehen dabei im Fokus. Diese Untersuchung ist wichtig, um meine wissenschaftlichen Beiträge im richtigen Kontext einordnen zu können. Als Resultat dieser Untersuchung wurden zwei interessante Anwendungsbereiche identifiziert: Quantified-Self und Smart Home Anwendungen.

Bei der sogenannten Quantified-Self Bewegung geht es um die Selbstvermessung mit Zahlen und Kurven. Dahinter steckt ein stark wachsender Markt, ein wenig Hype, aber auch viel Potenzial für auf mobilen Sensoren basierenden Anwendungen. Das Tracking von Aktivitäten wie Gehen, Laufen oder Radfahren basiert für gewöhnlich auf den in mobilen Geräten häufig vorhandenen Beschleunigungssensoren. Dabei kann es sich zum Beispiel um ein Smartphone oder eine Smartwatch handeln. Aber auch stationäre Übungen wie Kraftübungen sind für eine gesunde Lebensweise wichtig. Sie stärken vor allem die Koordination und Balance des Körpers. Solche Übungen können von einem einzigen Beschleunigungssensor nur unzureichend vermessen werden, da sie koordinierte Bewegungen mehrerer Gliedmaßen beinhalten. In dieser Arbeit stelle ich zwei alternative Sensortechnologien zur mobilen und berührungsfreien Vermessung solcher stationären Aktivitäten vor.

Die erste Anwendung basiert auf Ultraschallmessungen mit Hilfe der in gewöhnlichen Smartphones vorhandenen integrierten Hardware. Da moderne Smartphones zahlreiche integrierte Sensoren und die Rechenkapazitäten eines guten Arbeitscomputers besitzen, können Smartphones für die automatische Erkennung von menschlichen Aktivitäten genutzt werden. Dabei nutze ich das interne Mikrophon, um 20 kHz Ultraschallsignale zu versenden. Mit Hilfe der im Echo enthaltenen Doppler-Information extrahiere ich dann die charakteristischen Eigenschaften der verschiedenen Bewegungsabläufe. Smartphone Applikationen haben den Vorteil, dass sie mobil sind und keine zusätzliche Hardware benötigen.

Die zweite Anwendung ermögliche die berührungslose Vermessung dieser Ganzkörperaktivitäten durch kapazitive Sensorik. Ähnliche Anwendungen können mit Hilfe von Drucksensoren auf flexiblen Oberflächen realisiert werden. Die Nachteile sind dabei jedoch die mechanische Verformung sowie die notwendige direkte Berührung. Die kapazitive Sensorik ermöglicht eine verbesserte, berührungslose Interaktion. Die Reichweite wird von der Größe der Messelektrode bestimmt. In Relation zu der Anzahl der Messelektroden bietet ein solches System zudem eine höhere Genauigkeit.

Beim zweiten identifizierten Anwendungsbereich der Aktivitätserkennung durch Sensoren in einer intelligenten Umgebung handelt es sich um die Indoor-Lokalisierung. Solche Systeme liefern die genaue Position der Nutzer in Innenräumen. Mit Hilfe der Lokalisierung können Bewegungsprofile der Nutzer erstellt werden. Diese erlauben die Entwicklung komplexerer Systeme wie zum Beispiel assistierende Technologien für den Alltag, sicherheitsrelevante Anwendungen oder auch spezielle Anwendungen wie die Früherkennung von Demenzpatienten. Dabei stelle ich zwei Systeme vor, die unterschiedlich skalierbar sind. Die erste Anwendung basiert auf der Ausbreitung von Oberflächenvibrationen, die durch Schritte oder andere Gegenstände verursacht werden können. Dabei werden die entstehenden Vibrationsmuster in den Zeitsignalen genutzt, um Alltagsaktivitäten wie Schritte oder das Öffnen und Schließen von Schranktüren zu erkennen. Die Verwendung von Sensorarrays ermöglicht hierbei die Positionsbestimmung des Schallursprungs und fügt so dem Aktivitätsmuster noch eine Ortsinformation hinzu. Weiterhin wird untersucht, wie sich ein solches System mit minimalistischem Sensoraufbau realisieren lässt. Das zweite tag-freie System zur Innenraum-Lokalisierung basiert auf elektrostatischen Sensoren. Die Elektrostatik befasst sich mit der Verteilung elektrischer Ladungen und den elektrischen Feldern der geladenen Körper. Durch menschliche Bewegungen werden elektrische Ladungen verschoben und auf den Sensoren induziert. Elektrostatische Sensoren messen rein passiv und können dadurch sehr energieeffizient betrieben werden. Durch den gitterförmigen Aufbau und die Möglichkeit der Aufteilung in separate Untersysteme ist das Gesamtsystem auf unterschiedliche Raumgrößen skalierbar und lässt sich auf die vorhandene Raumgeometrie flexibel anpassen.

Im letzten Schritt beschäftige ich mich mit der Überführbarkeit von Prototypen zu realen Anwendungen. Diese stellt oft eine besondere Herausforderung dar. Die Problematik liegt oft darin begründet, dass sich das entwickelte Modell nicht an die reale Anwendung anpassen lässt. Es existieren große Unterschiede zwischen den während der Modellbildungsphase gesammelten Daten und den im realen Einsatz vorkommenden Eingangsdaten. Es existieren verschiedene Verfahren, die das Ziel haben diese Unterschiede zu minimieren. Ich habe anhand der hier vorgestellten Anwendungen untersucht, inwieweit diese Verfahren die Anpassbarkeit der Systemmodelle an reale Anwendungsszenarien verbessern können. Zudem wurde untersucht, wie sich die offline entwickelte Modelle auf online Plattform portieren ließe, die nur beschränkte Rechenkapazität besitzen. Diese stellte Anforderungen sowohl an das Modellkapazität als auch an die Verarbeitungsalgorithmen dar.

Zum Abschluss der Arbeit fasse ich die Erkenntnisse der behandelten Themen zu einer Schlussfolgerung zusammen. Des Weiteren gebe ich einen Ausblick über die daraus hervorgehenden interessanten zukünftigen Forschungsrichtungen, die ich im Rahmen meiner zukünftigen Arbeit weiter untersuchen möchte.

Acknowledgement

First of all, I would like to express my sincere thanks to my supervisor, Prof. Dr. Arjan Kuijper, for his patience and guidance through the whole process. His support has been invaluable. I benefited greatly from his broad scientific knowledge, his creative ideas, and his thorough reviews on my papers. His reviews are always constructive, valuable and improved the structure of my work. I would like to thank Prof. Dr. Dr. eh. Dieter W. Fellner and Prof. Dr. Kristof van Laerhoven for the valuable support and the agreement to evaluate this thesis.

I would like to express my appreciation to Florian Kirchbuchner and Dr. Andreas Braun. They are instrumental in defining the path of my research. For this, I am extremely grateful. I am also very grateful to Dr. Tobias Grosse-Puppendahl, who encouraged me to write my first scientific publication in my life. From the day on, I found the joy in writing papers and sharing my ideas with the scientific community. Throughout the years as a research scientist in Fraunhofer Institute for Computer Graphics Research IGD, I have been collaborating with many brilliant minds, such as: Julian von Wilmsdorff, Dr. Naser Damer, Dirk Siegmund. I am extremely thankful for the fruitful discussions and the scientific exchange of ideas. Many thanks go to my dear colleagues, Silvia, Daniel, Philipp, Fadi, Meiling and Olaf for providing such a nice working environments.

In the process of doing this work, I collaborated with many bright students. Their hard work and dedication, self-involvement, and the many in-depth discussions have inspired me a lot and made this work possible. Many thanks goes to my former bachelor and master thesis students: Jakob Karolus, Lennart Jarms, Matthias Ruben Mettel, Christian Stoll and Dinesh Vaithyalingam Gangatharan.

I would like to thank my family for the unconditioned support and encouragement throughout the whole process. Thanks to my husband Patrick Reichensperger for his patients and support in many stressful moments. Special thanks go to my parents (Fengjin Fu and Jingfang Xu) for their unconditional love and support. My father has inspired me with his wisdom, commitments, and determination to achieve the goal.

At last but not least, my gratitude goes to all of my friends who directly or indirectly helped me to complete this project. Finally, any omission in this acknowledgement does not mean lack of gratitude.

Contents

1. Introduction

Human activity recognition (HAR) is the automatic recognition of individual physical activities or group activities. HAR appeared to be the key research aspect in Human-Computer Interaction (HCI) over the last few decades. At the beginning of HAR research during the earlier 1990s [BBS14], researchers conducted the first feasibility studies with inertial body-worn sensors on action recognition. However, the choice of activity sets were rather constrained, arbitrary and less relevant to real-world applications at that time. It was in the earlier 2000s, when the important domains were identified that can benefit from the recent advances in the research of activity recognition, such as the industrial sector, office scenarios, the sports and entertainment sector, and healthcare. Understanding human actions in daily living enables application designers to build assisting smart home applications for elderly care [CNW11, CHN*12], safety applications with video surveillance [SBTM08], applications for Quantified-self [SCZ*14, KAY*18], or to associate physiological signals with emotions [KBK04] to build interactive applications.

Research fields and application areas of HAR are diverse. Today, the variety of sensor types are sky-rocketing. Sensors are all around us. One of the highest rates of growth of sensor deployment has been on the smart home domain. Miniaturized sensing devices are widespread. The distributed sensors build up an invisible wireless network connecting everything together. By the end of 2018, statics [HSA*16] stated that there were around 22 billion internet of things (IoT) connected devices used worldwide. Forecasts suggest that by the end of 2030, around 50 billion of IoT devices will be in use around the world. I will name four best examples of IoT applications. Ranging from smaller gadgets to large applications, the possibilities of IoT are infinite.

Smart home gadgets: Smart home gadgets aim at providing the inhabitants a more secure and convenient home experience. *Smart lock* is a device to mitigate the issue of misplacing a physical key. Access can be granted temporally or provided by using a smartphone. IoT thermostats allow us to adjust the temperature according to individual preference for a more granular control and energy saving aspect. *Smart mirror* in the bathroom can be used to display the weather condition, time, date, and other notifications from your smartphone to keep you up-to-date every morning.

Smart manufacturing: Manufacturing intends to benefit from the IoT in terms of cost saving. Improving automation, networking, and enhanced data analytics can prevent and detect possible issues in the process chain at an early stage. *Digital twins* are a copy of physical objects that are accurately simulated by the measurements from sensors. They aim at facilitating their owners ability to experiment on the asset and get a better understanding of the object. This also simplifies the production process and planing due to data-driven prognosis with a digital twin of a real physical object. Google glasses for example with augmented reality features can project manual instructions directly in the user's field of vision to speed up the construction process.

Smart farming: Another word for smart farming is precision farming. This aims at using digitization, modern machine learning tools and increased automation to make a direct impact on how the plants are nurtured and grown. BoniRob [RBD*09], developed by Bosch, is an automated robot that can distinguish between plants and weeds using sensors, algorithms from machine learning and image recognition. In case weeds are localized, they will be eradicated mechanically in order to let the plant grow freely. This reduces the needs for pesticides. Drones equipped with infrared and visible cameras are used to drive away wild animals.

Futuristic driver-less cars: The car picks you up and drops you off on your destination solely on its own might not be just a science fiction in the near future. Equipped with tons of sensors, cloud architecture, internet and more, these collected data can be used to build smart algorithms helping the car to perceive its environment and make the correct control decisions. Car-to-X communications enable a fully automated information exchange based on 5G technologies.

This thesis works with sensor-driven applications for HAR in smart environments. A smart environment [CD04] is an instrumented room or space using sensors and actuators to perceive the physical state or human activities within this space. In such a smart environment, users are able to seamlessly interact with each other. To create such an intelligent space, the ability to learn the knowledge about human activity, states and group dynamics from raw sensor inputs is of great importance. Sensors are devices that provide such ability and can help to detect and quantify physical aspects of the world around us. They can measure the intensity of light, translate the degree of heat into temperature, or turn mechanical pressure into a force quantity. I identify two use-cases that can be improved using novel advances in sensor technologies for a more convenient sensing and recognition: Quantified-self and indoor localization.

Quantified-self applications aim at building self-tracking tools to monitor the individual physical states. According to the definition of the Quantified-self community, it means self-knowledge through numbers. Encouraging people to regularly exercise is a well-researched topic in ubiquitous computing, especially using body-worn inertial sensors [KWM11a, LL13]. Tracking and recognizing the respective activities have successfully been implemented for various aerobic endurance training exercises, such as *walking, running*, or *cycling*. On the other hand, there is limited amount of research on the topic of recognizing more stationary exercises, such as strength-based training or stretching, without the use of wearable sensors. These are proven useful especially, as they prevent injuries and are essential for rehabilitation [U.S08]. Typically, these exercises are harder to track than *walking* or *running*, as they rely on coordinated movement of specific body parts. I investigate several sensor technologies to provide novel sensor solutions for this task.

Indoor positioning aims at providing location information to build intelligent services for smart building or smart home. The context of knowing the exact position of the inhabitant can be leveraged in a large number of novel application domains, such as health care, home care, anomaly monitoring or behavioral analysis. GPS technology is commonly used for outdoor positioning, but it is less efficient for performing positioning in indoor environment. Due to shading and multipath effects, the position information is imprecise and erroneous. Indoor positioning systems are either tag-based or tag-free systems. Typical tag-based systems are RFID-based [JLP06] or WiFi-based [AY09]. For tag-free systems, the user is not required to wear an identification tag to be localized. Sensing technologies for tag-free systems are for example capacitive [BHW11] or pressure-based [BHH*13]. In this thesis, I investigate sensing technologies beyond the existing ones to build a floor-based tag-free indoor positioning system on different scales. These proposed systems are less constrained than tag-based systems and more efficient compared to common tag-free systems.

Connecting the ever-increasing demand on new HAR applications and the ever-growing number of integrated sensors, there are new potential areas of application emerging that I want to deal with in the thesis. Understanding the role of sensors in the task of human activity recognition is thus an important research direction. Researches are not only restricted to use on-body sensors. The aim of HAR is now to enable human-centred applications and natural interactions in smart environments.

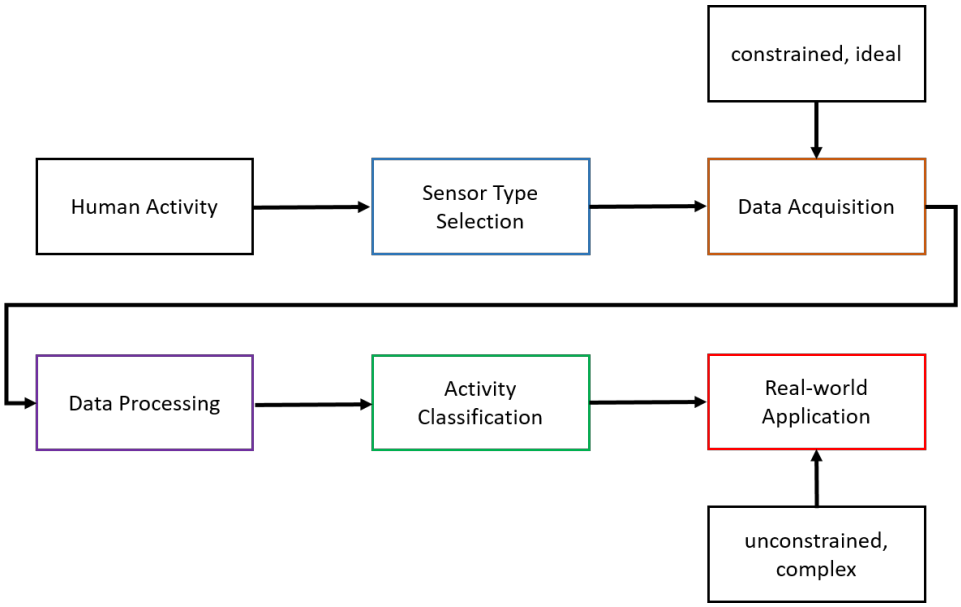


Figure 1.1.: Typical pipeline of an Activity Recognition Toolchain (ART) as a general-purpose framework for developing a sensor-driven application for human activity recognition task. The five colored blocks represent the five stages of the design steps for developing a sensor-driven application from prototypes to useful products.

1.1. How to setup a successful HAR application?

Designing sensor-driven applications for HAR task in a smart environment often poses a set of open design questions before the implementation phase. Common questions are like for example: *what are the targeted use-case, what type of sensor should be used, how should the system design look like to fulfill the design requirements*, and a lot more. These considerations lead to the main research question of this thesis regarding *how to setup a successful HAR application*?

In order to structure these common questions, I propose a typical pipeline of an Activity Recognition Toolchain (ART) as a general-purpose framework for developing a sensor-driven application for HAR. Process pipelines for sensor system design are common. They could be ranged from fairly simple to more complex compared to the proposed one. Here, I adopted the general-purpose framework according to the best practice model related to common knowledge gathered from my work experience with these sensor applications. This pipeline is illustrated in Figure 1.1. The first stage focuses on **sensor type selection** in regard to the physical activity to measure. The second stage, **data acquisition** deals with the process of preparing the setup and collecting data. The third step, **data processing** involves the data preparation for further processing or system modelling. The fourth step concerns with the topic of data-driven modelling, which mainly focuses on the **activity classification** task using supervised learning methods based on labeled data from sensor input. Finally, the last stage is to deploy the trained model to **real-world application** without losing model performance.

I will first motivate the structure of the proposed framework using a sample application. Then, according to this general-purpose framework, I relate in detail the five identified main research fields. These five research fields are indicated by colored frames in Figure 1.1. According to the research fields, I pose the individual research

questions resulted from the specific challenges in each of the research field. My solutions and contributions towards these research questions are shortly summarized in Section 1.2.

Since the aerobic endurance training, such as walking, running and cycling have been well studied using wearable designs, I dedicate the sample application with the preliminary goal to detect strength-based physical activities to motivate the components in the proposed framework. These activities require novel sensor solutions to solve the challenges they face as these activities are often not well recognized with a single wrist-worn solution.

According to the defined use-case, we first contemplate what needs to be detected. The prerequisite of whole-body exercise recognition is to detect different movements from all body parts. For periodic movements we further want to determine the repetition of each exercise. Counting the number of exercise contributes to building personal statics and keeping track of self-development. Other design questions are such as:

- What are the different base postures to composite a more complex exercise?
- Are these exercises mostly in a standing or lying position?
- Should the application be for indoor or outdoor environment?
- Should the system be designed for on-body sensing or remote sensing?
- Is the prototype determined for single person or multiple person use-case?

among others. These are examples of basic questions that should be answered prior to the application design phase.

1.1.1. Sensor type selection

Considering the design requirements previously posed, I first introduce relevant aspects to be considered for the task of sensor type selection. Those aspects can be formulated as follows:

- Which physical capability does a sensor possess to measure the required body movements?
- Should the interaction be implicit or rather explicit?
- How can privacy be guaranteed?

With respect to these relevant aspects, I come up with several possible sensor types for this task. First, image-based measurements using camera in visual spectrum provide rich and detailed context information. This system has very low impact on user and thus enables them to behave naturally. However, vision-based system suffers from occlusion and changing illumination problems. Beyond the technical challenges, camera system also raises privacy issues in public and private sectors.

Second, wearable sensing devices using acceleration sensor are commonly used to track motion information. Many miniaturized devices, such as smartwatches or smartphones nowadays, already have built-in acceleration sensors on the device itself. Such an integrated device poses low restrictions on the user. However it requires a correct "handling" of the sensing device. The placement and a correct usage are directly related to the performance. For certain exercises, where some body parts are dormant, the sensor placed on such locations can not provide any useful motion information.

Third, today's smartphone is a cornucopia of information. The huge variety of sensors in today's mobile phones makes these devices a prime target for HAR tasks. However, using the smartphone as a sonar device to measure movement with the built-in hardware is only leveraged in recent years. The advantage of this sensing modality is that it allows remote sensing for human activities. The privacy is preserved by avoiding the visual input and it is resistant to illumination changes. Negative aspects however, conditioned on the physical sensing principle, such as low detection range and the problem of occlusion pose further limitation at the design phase and make the system suitable for mainly near-range application.

Finally, the proximity sensing prospects of capacitive and pressure sensing can be leveraged for this use-case. Capacitive and pressure resistive sensors are widely used in the current touch screen applications. However, they can also be leveraged to build interactive appliances due to its flexibility in the usage. Capacitive proximity sensing allows remote sensing up to 15 cm. These system preserves privacy, while only using multivariate time series modulated by human actions. The drawback of capacitive system is due to its limited detection range and error-proneness towards environmental noise. In contrast to the other named sensing modalities, this sensor type requires additional hardware design.

In summary, there exists multiple possible sensing modalities, which are equally well suited to solve the exercise recognition task. The set of previously posed design questions can help us limit our choices on sensor selection. This open set of questions leads me to my first research question:

RQ1: Which sensor category has to be applied under which conditions?

1.1.2. Data acquisition

After the sensor selection for this task is done, we assume that we aim at developing the application in an offline fashion first. The next step is intuitively to think about the storage option and tools to record the exercises. In case of camera systems, smartwatches, or smartphones, these installations already have built-in storage available in most cases. For external hardware designs, such as for application with proximity sensors, additional storage is required to record the sensor values. Hardware requirements towards recorded signals are further conditioned on diverse aspects, such as sensitivity, signal-to-noise ratio, sampling frequency, synchronization and much more.

For classification tasks, we need to additionally consider the issue of labelling to be integrated in the data recording tool. How much data is necessary? How much data is required to model the data diversity? In order to model the user diversity, we need to collect data from participants with different body shapes and different degree of affinity towards sport exercises. Finally, it should be noted, that the data acquisition phase is often under controlled setup which differs from the real-world setup.

Therefore, these questions should be posed in the data acquisition stage and lead us to the second research question:

RQ2: What has to be considered for data acquisition with specific sensor technology?

1.1.3. Data processing

In this stage, data are recorded and labelled for the classification task. Now, we have to segment the data in time with respect to the requisite target exercise. The segmentation is closely related to speed or duration of different exercises. Improper segmentation causes difficulty to the classification task. Making the time window too short, we miss the overall structure of the exercise, while making it too large, we would incorporate more noise into the useful signal. Thus, the duration of a segmentation time window is guided by the duration of the diverse exercise movements.

Signal quality improvement is another essential task in the data processing stage. It aims at increasing the signal-to-noise ratio by removing the outliers and handling the invalid and missing data problems. Beyond the basic data handling methods, data should be standardized in prior to properly train certain machine learning models. However, it is an open question of how much processing is required. Over processing generally leads to performance drop in the later classification stage.

Data-driven modelling is strongly dependent on the underlying training data. The quality of the data is important for building a good performing model. Thus, this stage aims at improving the quality of data. Data processing is often a design choice by domain experts. This leads us to our next research question:

RQ3: What degree of data processing is sufficient without affecting the performance of the data modelling?

1.1.4. Activity classification

This stage of activity classification aims at constructing the appropriate model or model architecture to solve the given classification problem. This stage is strongly correlated to the processed data. Now, we can assume that the signals are clean and already well pre-processed. For multivariate time series, in case of acceleration data or capacitive proximity data, sequence models can be leveraged to model the temporary patterns for certain exercises. For image-like inputs, in case of visual output of a camera system or the two dimensional Doppler spectrum from audio signals, one can use spatial patterns to extract discriminative features from exercises.

In the activity classification stage, we deal with the challenge of constructing the right model architecture or model capacity for the given task. We aim at extracting useful relationships among the features and learning the correct representations. This leads us to the following research question:

RQ4: Which architecture or model has to be used for certain sensor application of HAR?

1.1.5. Real-world portability

From the previous stage, we have now a well trained model on classifying the exercise activities. The last step is to deploy the model to a real-world application. I have identified at least three difficulties: real-world data, online application and scalability.

Real-world data contains way more variability compared to collected training data under controlled setup. This is because, the target group in this case is more diverse and the data collection environment could differ from the controlled setup. The step of using the trained model on real-world data is to adopt knowledge from controlled dataset to uncontrolled dataset. In order to achieve a successful model adaptation, it is crucial to improve the generalization ability of the trained model.

Online application requires the adaptation of trained offline-model to be applicable for online processing. For embedded hardware, the computing power is limited. How to make sure, that the processing is adapted to those embedded hardware is thus a very important design issue. In addition, if instant feedback is required in certain use-cases, a real-time response is needed. The real-time requirement also limits the processing capability.

Scalability is considered with respect to the model or hardware setup. How easy is it to extend the model, such as adding new exercises? How easy is it to keep human operators in the loop for labeling hard samples in order to improve the model performance? Regarding the hardware setup, how can we scale the sensor setup?

The complexity and diversity in real-world data makes them challenging to handle. The next research question therefore deals with the model adaptation on real-world data.

RQ5: How to overcome the gap between constrained development data and the more complex real-world data with the scope on time series for HAR applications?

Standard procedure for developing a sensor application for activity recognition task is to implement the algorithm in an offline fashion by exploring with a fixed development set of data. After the model is successfully optimized on the development data, the last step is to deploy it to a live operation. Herein, we often face the challenge of limited target platform processing capability. This leads us to the final research question in this thesis:

RQ6: How to scale model complexity used in real-time applications?

1.2. Research challenges and contributions

Derived from the main research question of *how to setup a successful HAR system*, I proposed a general framework in Figure 1.1 based on common knowledge deduced from my work experience. Resulting from the framework, I further introduced the following sub research questions, this thesis is dealing with. In this section, I introduce the main contributions related to the individual research questions, this thesis is build upon.

Research question 1 *Which **sensor category** has to be applied under which conditions?*

Sensors used in HAR applications are manifold. Although several surveys have been conducted for HAR with specific sensor categories, such as surveys on acceleration-based [LL13, SAMM19], radar-based [LHJ19], radio-based HAR [WZ15] and camera-based HAR [KTL*13], these are all focusing on single sensor technology based applications for a subdomain of HAR. A thorough comparison across these sensing technique categories with a focus on the sensor advantages and disadvantages in specific tasks is still lacking. Other surveys focus on algorithm-based methods, e.g. recent advances made in deep learning [WCH*19, RRR18] and transfer learning [CFK13] applied in the domain of HAR. Therefore, I propose a sensor-driven categorization for HAR tasks in this thesis and surveyed sensor categories in the domain of HAR with respect to the sampled physical properties, including a detailed comparison across sensor categories. I further identify the limitations with respect to the hardware and software characteristics of each sensor category and draw comparisons conforming to the benchmark features retrieved from the research works introduced in this survey. Finally, I provide some guidelines and some intuition with respect to the posed research question.

This answer provided in chapter 2 is based on the survey by **Biying Fu**, Naser Damer, Florian Kirchbuchner, Arjan Kuijper: **Sensing Technology for Human Activity Recognition: a Comprehensive Survey.** IEEE Access (Volume: 8) 2020: 83719-83820. This survey further provides some insight to position the work of this thesis.

Research question 2 *What has to be considered for **data acquisition** with specific sensor technology?*

The data acquisition process for HAR tasks are most expensive and tedious. Researchers [UNH08, WGH07a], who aim to design multisensor systems to recognize activities of daily living, often have to construct extra spaces such as a laboratory environment simulating real living spaces to collect data. This makes data collected under such conditions differ from data collected in the users usual living conditions. In addition for supervised learning tasks, extensive manual labeling from domain experts or manual operators is required. Although, this manual labeling is prone to label errors. How to reduce this effort is thus an important research topic. In this thesis, I explored both options of labeling: labeling through an instructor and labeling through the user himself. Depending on the complexity of the underlying task, one approach is more beneficial than the other.

Related to the sensor specification and characteristics, parameter selection such as sampling frequency, calibration and value range is part of this design phase. I examine these parameters by exploring with configured systems containing fixed hardware specifications and self-designed systems with adjustable parameter ranges. Fixed hardware specification allows the freedom of parameter setting to a certain degree, while self-designed systems offer more freedom in this respect however with the increased cost of external hardware design and less generalizability for customized design. The data acquisition is to be performed under similar conditions as in the real targeted application scenario. This minimizes the difference between the development and the real-world data.

Research question 3 *What degree of **data processing** is sufficient without affecting the performance of the data modelling?*

Beyond the data segmentation, data imputation, and signal quality improvement, the feature extraction is the next processing step. The feature extraction is a critical step of extracting discriminative patterns from segmented data samples. This can be performed mostly in two different ways, which is either constructed with prior knowledge or automatically extracted according to task specific requirements. Traditional approaches often rely on feature engineering with handcrafted features according to domain experts' knowledge.

Referring to time signals, three feature domains are commonly used. They are either features from: time domain, frequency domain and time-frequency domain. In the time domain, we consider features mainly from the pure signal appearance, such as zero-crossing or other amplitude related features. In the frequency domain, Fast Fourier Transformation (FFT) [RKH10] is often applied to get the spectral information of the signal with respect to certain frequency bins. In the time-frequency domain, we are interested in the spectral distribution of the signal over discrete time steps. This can be achieved by using short time Fourier transformation (STFT) [Grö01].

Referring to image data, handcrafting features include object recognition [RDGF16], object segmentation [HGDG17a] and extracting statistical [DP16] or structural [MNR92] features on the pixel level. These features are heavily dependent on handcrafted heuristics from design experts. This imposes a strong restriction on the designed system in this domain and makes the transfer-ability to another domain more difficult. Therefore those systems are most suitable for recognizing simple tasks. For recognizing complex tasks, more powerful models are required.

Modern approaches have a shift of focus towards end-to-end learning, where the learning objective is to optimize the classification accuracy by integrating the feature extraction stage directly into the training network without the need to include any prior knowledge. This blurs the boundary between data processing and modeling.

Addressing this research question, I compare conventional feature extraction methods with respect to model-based automatic feature extraction methods using the developed application prototypes in this thesis. Grounded on these investigated results, I formulate common basic processing methods without restricting the model performance in the next step.

Research question 4 *Which architecture or model has to be used for **activity classification** in certain sensor application of HAR?*

Depending on the given problem, we should first decide if it is a regression problem or a classification problem. In this thesis, I focus mainly on the classification problem. According to the problem setting, most learning algorithms are either discriminative or generative. Generative models, such as Hidden Markov model [RJ86], Naive Bayesian [MVPEL18] or Gaussian Mixture Model [PK13], work well on dataset with few labels, while discriminative models, such as Support Vector Machine [GCC*19], simple Feedforward Neural Network [SA19] or k-Nearest Neighbours [GSC*17], work best on labeled dataset. Generative models aim to build a probabilistic model in compliance with the underlying data. It can be trained in a supervised or an unsupervised way. Discriminative models aim to find the optimal decision boundary separating samples into their proper classes. The model has to be trained in a supervised way and heavily relies on the amount of training data. The architecture, hyperparameters design and the model capacity are strongly dependent on the given data distribution and are therefore task specific.

Using the sensor-driven applications developed in this thesis, I compare the suitability of different architectures and models with the primary goal of improving the classification results on specific use-cases. By gathering

knowledge from the diverse applications, I provide common guidelines or best practice model related to the underlying sensor setup.

The solutions to research questions 2 to 4 are derived from several publications and summarized in Chapter 3. These research questions are considered in combination and are targeted by the individual application design and its modification to improve the performance. The contributions are mainly build upon the following four publications:

- **Biying Fu**, Jakob Karolus, Tobias Grosse-Puppendahl, Jonathan Hermann, Arjan Kuijper: **Opportunities for activity recognition using ultrasound Doppler sensing on unmodified mobile phones.** iWOAR 2015: 8:1-8:10. The work aims to show the feasibility of using a commercial mobile phone to detect human activities in general.
- **Biying Fu**, Florian Kirchbuchner, Arjan Kuijper, Andreas Braun, Dinesh Vaithyalingam Gangatharan: **Fitness Activity Recognition on Smartphones Using Doppler Measurements.** Informatics 5(2): 24(2018). Building upon the exploratory study, this work aims at detecting three distinctive sport exercises.
- **Biying Fu**, Florian Kirchbuchner, Arjan Kuijper: **Unconstrained Workout Activity Recognition on Unmodified Commercial off-the-shelf Smartphones.** PETRA 2020: 20:1-20:10. Extending the previous work, this work intends to recognize a more complex and realistic set of workout exercises.
- **Biying Fu**, Lennart Jarms, Florian Kirchbuchner, Arjan Kuijper: **ExerTrack - towards smart surfaces to track exercises.** Technologies 2020, 8(1), 17. Using customized hardware prototype, this work leverages multiple capacitive proximity sensors to focus on the same set of workout exercises in order to make a fair comparison across sensor categories.

Research question 5 *How to overcome the gap between constrained development data and the more complex* ***real-world data*** *with the scope on time series for HAR applications?*

Researchers and application designers often face the problem that the performance drops by applying a well trained classification model on real-world dataset. The reason is the inherent difference between the development set and the real-world dataset. This could due to the variations induced by the user or the environment. User-induced variation is due to the complexity in human actions. This term is called the user-diversity. Reducing this issue is more challenging, as it is inapplicable to include all diversities within the training dataset. On the other hand, the environment induced variation is easier to mitigate, in case it is a constant term and is not time dependent. Then, removing this constant noise term can be considered as mitigating a systematic error.

In this thesis, we investigate several approaches to improve the model adaptability with time series for unseen test data encountered in real-world applications. Being aware of the complexity in real-world data enables us to build robust systems without overfitting the model to data with bias. I address this research question both from the data space and the feature space. By leveraging the data augmentation techniques for time series, I aim at solving the problem in the data space, while the individual finetuning methods focus on a metric-based learning approach in the features space.

This research question is addressed in Chapter 4 and the main contributions are concentrated in the following two publications:

- **Biying Fu**, Florian Kirchbuchner, Arjan Kuijper: **Data Augmentation for Time Series: Traditional vs Generative Models on Capacitive Proximity Time Series.** PETRA 2020: 16:1-16:10. This work aims at putting regularization on the model by applying the data augmentation technique.
- **Biying Fu**, Florian Kirchbuchner, Arjan Kuijper: **Generalization of Fitness Exercise Recognition from Doppler Measurements by Domain-adaption and Few-Shot Learning** (accepted in 25^{th} *International Conference on Pattern Recognition* (2020), Workshop on Deep Learning for Human-Centric Activity Un-

derstanding). This work intends to target the challenge of data diversity and to improve the model generalization by using domain adaptation and few shot learning methods with few labeled samples only.

Research question 6 *How to scale model complexity used in **real-time applications**?*

The final production system with its available resource could pose limitations on the developed model. The production system can be either a server in the cloud, a working desktop PC, or an embedded device. A small model with moderate model capacity is more likely to be directly running on an embedded device with limited computational power. Recently, there exist approaches to convert a large, more powerful model to a smaller one without losing much accuracy by leveraging the approach called knowledge distillation from Hinton [HVD15].

Addressing this research question, I modulate the model capacity to fit it on standalone devices with restricted processing capability. Other solutions, such as reducing the overhead of generating handcrafted features or limiting the feature dimensions and mitigating the feature correlations with dimensionality reduction techniques are explored. The applications developed in Chapter 3 and 6 are designed with the goal to run the application on edge devices in the real-world scenario to protect user privacy.

The answer to this research question is summarized in Chapter 5 and derived in the following contributions:

- I demonstrate the setup of deploying a pre-trained model on a Raspberry Pi 3. It partially includes the result from the master thesis of Lennart Jarms with the title **CapMat for sport exercise recognition and tracking** supervised by me.
- **Biying Fu**, Tobias Grosse-Puppendahl, Arjan Kuijper: **A gesture recognition method for proximity-sensing surfaces in smart environments.** HCI (21) 2015: 163-173. It demonstrates the possibility of performing the mid-air gesture recognition using a simplified model-based approach on a standalone device with limited resources.

Main research question *How to setup a successful HAR application?*

After targeting the individual research questions in the separate chapters of this thesis, in Chapter 6, I introduce two stationary applications deployed in smart environments following the findings in the individual steps of the proposed framework in Figure 1.1. This chapter aims at providing an overview of applying the previous findings to answer the main research question about setup a successful HAR system from system design to a working prototype.

The main contributions are detailed in the two following publications:

- **Biying Fu**, Matthias Ruben Mettel, Florian Kirchbuchner, Andreas Braun, Arjan Kuijper: **Surface Acoustic Arrays to Analyze Human Activities in Smart Environments.** AmI 2018: 115-130. This work deals with extracting temporal and structural patterns from vibration signals to recognize fine-grained activities of daily living.
- **Biying Fu**, Florian Kirchbuchner, Julian von Wilmsdorff, Tobias Grosse-Puppendahl, Andreas Braun, Arjan Kuijper: **Performing indoor localization with electric potential sensing.** J. Ambient Intell. Humaniz. Comput. 10(2): 731-746 (2019). This work aims at building a scalable indoor localization system to provide accurate indoor positioning for smart control or safety assisting appliances.

1.3. Structure of this thesis

After introducing and motivating to the topic of HAR, I posed the research questions this thesis is focused on derived from the main research question of *how to setup a successful HAR application*. To address this research

question I further proposed the generalized framework in Figure 1.1. The structure of this thesis is strongly related to the individual components within the framework and is organized as follows:

Chapter 2, *Related Work* aims to solve the first research question *RQ1* regarding the sensor selection according to task. This chapter is a collection of the recent research works grouped by sensor categorization in the domain of HAR. The first part of this chapter presents our sensor categorization scheme according to the physical entity they measure and revises the most prominent works utilizing these sensor categories in the domain of HAR. In the second part, I provide a detailed discussion of public available databases intend to help developing applications in this research domain with the corresponding sensor categories. I then introduce the common evaluation metrics used in the literature to evaluate and compare the performance of the developed algorithms and systems. Finally, with respect to the identified hardware and software limitations, I provide the application designer with some insight and guidelines into selecting the appropriate sensor categories.

Chapter 3 engages at developing applications contributing at missing sensor applications in HAR. The design process aims at solving the research questions *RQ2* to *RQ4* in combination. These questions deal with challenges and issues concerning data acquisition, data processing and the final modeling.

The *mobile applications* in Chapter 3 fill the gap of deploying other suitable sensor technologies for Quantified-self applications beyond the existing technologies. I leverage two specific sensor categories, ultrasonic sensing with an unmodified off-the-shelf smartphone and capacitive proximity sensing with sparse implementation for real-time usage. I describe the application design following the framework previously introduced in Figure 1.1 and detail our design process in the individual application section. Before designing the application with mobile devices to recognize whole-body exercises, a thorough experiment is first performed on the physical sensing characteristics and the feasibility of the mobile device used as an ultrasound sensor. In regard to the experimental outputs, I successively increase the level of complexity to build a more sophisticated application to determine a set of complex, more realistic and diverse human activities. Addressing the deformation issue of a similar pressure-based textile application for physical exercise recognition, I propose to enhance a consumer yoga mat with capacitive proximity sensors to discriminate fine-grained whole-body exercises. In contrast to pressure-based sensing, capacitive proximity sensors enhance the sensing modality by sensing objects up to 15 cm distance without enforcing a touch interaction. The electrode material of copper plates increase the robustness of the system while still remaining portable.

Chapter 4, *Real-world data* aims at contributing towards the research question *RQ5*. The objective of *RQ5* is to find methods or algorithms to overcome the gap between the inherent difference in the development data and the more diverse real-world data. I investigate two different approaches to increase the model generalization ability on real-world data not present in the earlier training stage. The first method works with time series augmentation to increase the variability in the training data domain and to decrease the variance in the performance on the test set. The second method focuses on individual finetuning on human activity data. Using metric-based learning method, I intend to minimize the gap in the feature embedding space between the individual data, hardware and different acquisition environments. The methods investigated in this chapter is evaluated on collected dataset with applications designed in Chapter 3 from the mobile applications.

The objective of *RQ6* deals with deploying the offline model to an *online application*. To target this quest, I intend to scale model capacity and reduce processing efforts to fit on devices with limiting processing resources. Addressing the real-time applicability of the designed models, I demonstrate two possible solutions in Chapter 5. The first solution is detailed by showing how to deploy the enhanced yoga mat designed in Chapter 3 to a Raspberry Pi 3 for real-time usage. The second example illustrates a simplified model-based design for mid-air gesture recognition with capacitive proximity sensors on a standalone device.

Chapter 6, *Stationary applications* are installed systems that can be used to ubiquitously sense the human activities in a smart environment. This chapter is used to show instances demonstrating the individual design

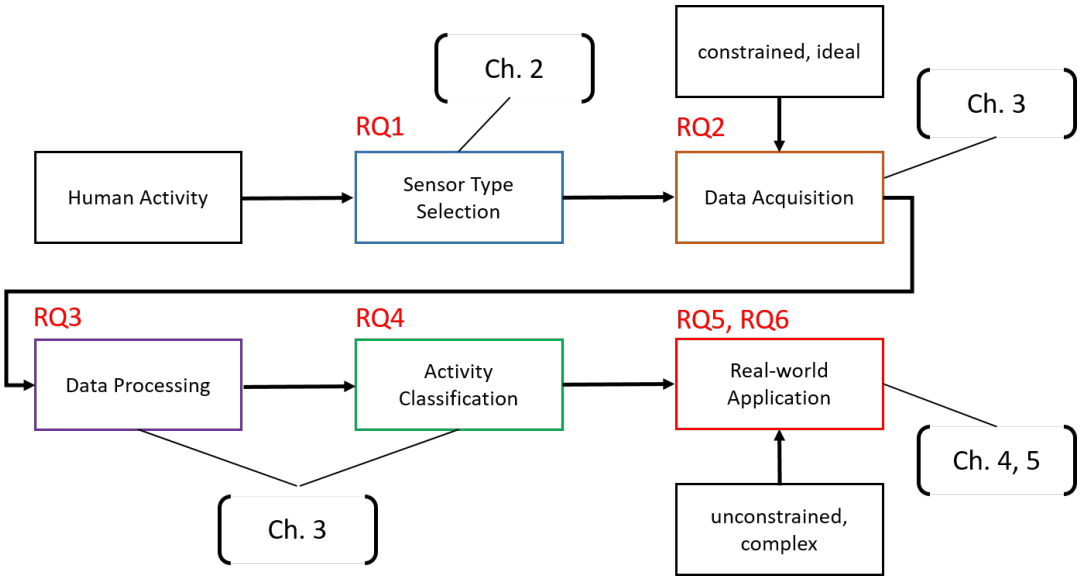


Figure 1.2.: Typical pipeline of an Activity Recognition Toolchain (ART) as a general-purpose framework for developing a sensor-driven application for human activity recognition task. The thesis chapters are linked to the corresponding research questions and individual framework components. Chapter 6 is used to show-case the overall process pipeline. Chapter 7 concludes the thesis contribution relate to the RQs and provides other future research directions.

choices in correspondence to the main research question *how to setup a successful HAR application* by following the general-purpose framework. The first application leverages the surface vibration caused by human motion or induced by acoustic events to detect a fine-grained set of activities of daily living. I aim at designing a tag-free surface acoustic array for analyzing human activities in smart environments, such as a real-time fall detection among others, and further investigating the minimum sensor setup for accurate detection with minimum computation effort. The second application is a floor-based indoor positioning system using electrostatic measurements. The sensed electrical signal is caused by body electric charge modulation via dynamic body motion. Such a system with grid-based layout and extendable system capacity is scalable to different room geometries and sizes. The primary goal is to build a low cost sensor system with high sensitivity and precision.

Chapter 7, *Conclusion and future work* summarizes the contribution of this thesis related to the posed research questions and provides some future research directions in this exciting research field of HAR with sensor applications.

Figure 1.2 links the chapters and the research questions to the appropriate stages within our proposed processing pipeline for designing a sensor-driven application. This framework is previously presented in Figure 1.1.

2. Related work

In this chapter, I conducted a sensor-driven survey that considers all sensor categories in the domain of human activity recognition (HAR) with respect to the sampled physical properties, including a detailed comparison across sensor categories. This chapter is partially based on the survey published in [FDKK20b]. This survey is used as a guideline to identify possible contributions in this field of HAR and aims at answering the research question 1 regarding sensor type selection with respect to the targeted HAR task.

"In physical science the first essential step in the direction of learning any subject is to find principles of numerical reckoning and practicable methods for measuring some quality connected with it. I often say that when you can measure what you are speaking about, and express it in numbers, you know something about it; but when you cannot measure it, when you cannot express it in numbers, your knowledge is of a meagre and unsatisfactory kind" by Lord Kelvin (William Thomson) [Die16]. Sensors are devices that can help to detect and quantify physical aspects of the world around us. Sensors are all around us. One of the highest rates of growth of sensor deployment have been in the automotive sector. A modern automobile is equipped with an average of 60 to 100 sensing devices with a rising trend mainly for functional aspects, such as the engine operation, brakes, safety, or emission controls [RI05]. With the growing trend of smart vehicles, the demand on more sensing units is expected. Also in the smart home domain, miniaturized sensing devices are widespread. The distributed sensors build up an invisible wireless network connecting everything together.

In order to facilitate a sensor comparison and obtaining a comprehensive overview of the sensing technology, researchers try to categorize them into different categories. Sensor classification scheme can range in its complexity. Simple general schemes commonly conclude three sensor categories based on the nature of the sensed property (physical, chemical, and biological) [Whi87]. However, a more complex categorization is often required when addressing distinguished applications. This work focus on the sensing technology deployed in academic research and consumer products for HAR. To build our sensor categorization within this field, we adopt the classification scheme proposed by White [Whi87]. This scheme is accredited to be more flexible and intermediate in complexity. It is according to the measurands or physical entity that a sensor actually senses such as temperature, light intensity, or mechanical stress. We present a first look at our categorization scheme in Figure 2.1, where we show the first level categorization based on the physical quantities followed by common sensor types utilized to measure this appropriate physical quantity.

We categorize sensors according to its physical properties to adjudge its affiliation to sub-domains of HAR. Tasks may differ, but the sensor physical characteristics remain. The appropriate sensor category to use is left as a design choice to the application designers. Using this survey, the application designers should be able to consider the appropriate sensor category with respect to specific task. This survey provides useful insight for researchers and developers in the HAR domain and provides a summary of existing works, including insight into the current and future research directions.

This chapter is organized as follows: we first present our sensor categorization scheme according to the physical entity they measure and revise the most prominent works utilizing these sensor categories in the domain of HAR. We then provide a detailed discussion of public available databases intend to help practitioners developing applications in this research domain with the corresponding sensor categories. Further, we present the common evaluation metrics used in the literature to evaluate and compare the performance of the developed algorithms and

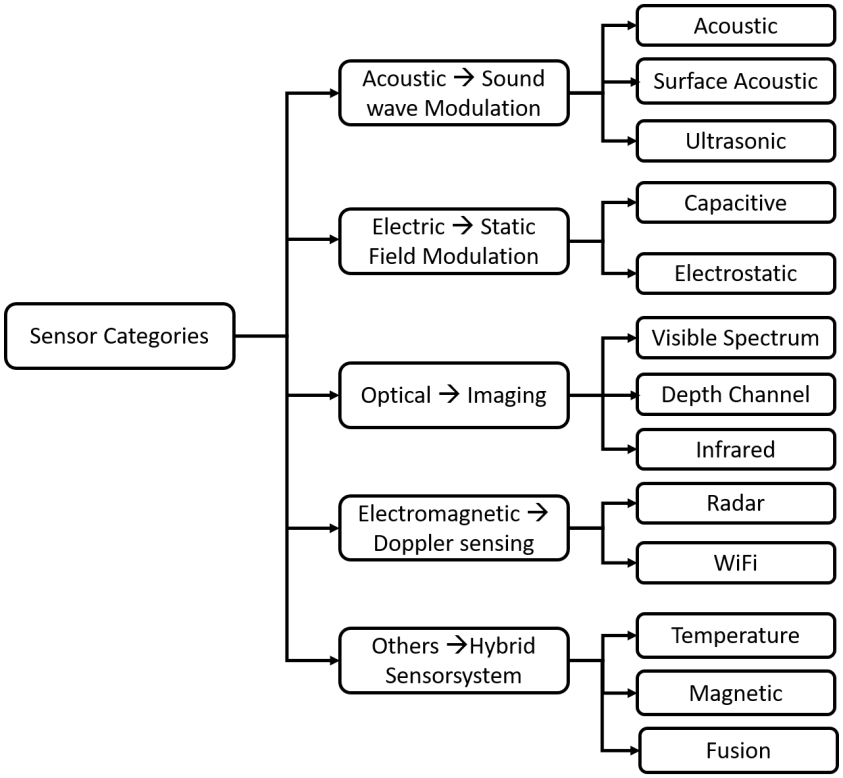


Figure 2.1.: The sensor categorization for HAR as presented in this work and based (at the first categorization level) on the work by White [Whi87]. We further extended this definition to include the measuring methods, commonly used in the domain of HAR.

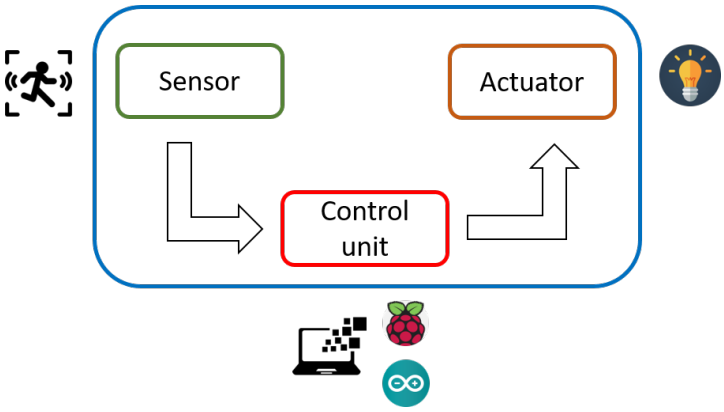


Figure 2.2.: A sensor plays an essential part in an automated system. It senses certain properties of the environment and convert it to electric input feed to the central control unit. The control unit makes a decision in line with the digital input data and makes the actuator act upon this decision.

systems. In addition, we give a thorough discussion on the hardware and software limitations we identified for each sensor category resulted from the literature research conducted within this chapter. Finally, we summarize this chapter in respect to the research question 1 and provide useful insight into possible solutions towards the mentioned challenges and offer an overview on current and upcoming future research directions in the domain of HAR with sensory data.

2.1. Sensors

A sensor is in general a converter that turns a physical quantity into electric values to be perceived by a digital system. Its output changes according to the change of physical properties on the input side. Sensors integrated in smart environments can either unobtrusively perceive the environment or be directly interacted with. Sensors that tend to sense the natural human intention without direct interaction can be used to design implicit interaction interfaces. Sensors that expect the user to initiate a direct interaction is used to design explicit interaction interface. To choose the appropriate sensor type to design the corresponding interface requires a clear sensor classification. Here we divide the sensor types into acoustic, electric, mechanical, optical, and electromagnetic and introduce its related physical sensing properties.

Typically, a sensor works in close collaboration with actuators and control unit to build the full cycle of an automated system, as illustrated in Figure 2.2. What a sensor measures will be interpreted by a logic unit, which is the decision making layer and leads to certain action triggered by it. An actuator acts the correct response according to the measured entity from the sensor.

In this survey, we only focus on the sensing part and portray all possible physical entities, which are commonly used to perform HAR. The miniaturization of sensing devices and the cheap production cost make smart sensing devices widespread in the smart home domain with an aim to simplify our everyday life. Voice assistants such as *Alexa, Siri, Cortana* and more [Hoy18] can listen to our voice command and control the lightening or other smart appliances. For human-centred designs, it requires to understand the human actions performed. Sensors can make the link between the human actions and the interpretation unit. The same human action can be measured by various sensor types, but the pool of actions is wide, which makes an action-based comparison more

difficult. Therefore, in order to make a more easily comparison across sensors, we make the sensor classification based on the physical measures and provide related applications with this type of sensor used in the subdomains of human activity recognition, such as indoor localization [STS*13, VMV09, FKvW*18, GPDH*16, BHH*13, TKY*16], home behavior analysis [SWvHG11, TN06, ARK*06, CSZ*16, JHJP08, UNH08, SA19], Quantified-self [GSC*17, SCZ*14, KAY*18], gestures, postures recognition [QHX*14, NITG16, PBRW*08, CMPT12, SBQK05, LGK*16, WSL*16, AYH15] and sensing of physiological signals [NGW15, PBRW*08, RLBL*18, ZWX*19, AK13].

Physical quantities, such as sound, light and pressure can be measured by acoustic sensors, optical sensors and pressure sensitive sensors. In the following sub-sections, we present some detailed works with regard to the sensor categorization given in Figure 2.1. The common structure for each sensor category is organized as follows:

1. introduce the physical sensing principle,
2. survey the most prominent research works that utilizes the questioned sensing category in activity recognition,
3. conclude and discuss the utilization of the sensing technology, including the advantages and disadvantages within the application domain,
4. a summery of the discussed works with a clear table-structured presentation of the main take-home-messages.

2.1.1. Acoustic

Acoustic sensors can measure mechanical or acoustic waves traveled through concrete materials. The transmission speed is easily affected by the different material properties over the propagation path in the transmission channel. Mechanical waves traveled through solid materials, can be detected by a surface acoustic sensor. Typical representatives of a surface acoustic sensor are built with piezo-electrical elements. These sensors are mostly operated in passive mode. Seismograph is a passive sensor, which could be used to measure the vibrations on the ground surface caused by a step signal. Passive sensors are compact, cost efficient, easy to fabricate, and have a high performance, among other advantages. However seismic sensors need a robust ground coupling to detect the vibrations traveled through the surface. The better the coupling, the better will be the signal-to-noise ratio of the received signals. Active acoustic sensor can measure sound waves transmitted through the air channel. These sensors can generate an electric signal, which will be converted to mechanical oscillation by using a membrane to set the air around the transducer into motion. This mechanical wave will be modulated by the object or obstacles close to the sensor and the back reflection is sampled by an analogue digital converter (ADC) converting the echo modulation back to electric signal. In this subsection we will discuss three main categories of this sensing technology: active acoustic, surface acoustic, and ultrasonic sensors. This subsection will later include an overall discussion of the technology and a final conclusion.

Active acoustic sensors Sound events such as clapping, coughing, laughing and yawning, besides natural speech languages may carry additional information for perceptual aware systems. Schroeder [SWvHG11] proposed using a microphone to detect four acoustic events (coughing, knocking, clapping and phone bell). Several signal processing steps and template matching from the frequency spectral domain are necessary to extract useful patterns to train the SVM classifier. Temko [TN06] focused on identifying 16 types of meeting room acoustic events, such as *chair moving, door slam, coughing, laughing, etc.*. Their source of sound samples are acquired both from the public database, such as **RWCP** [NHA*00] , **ShATR** [VYK13] database and the world wide web.

However the class distributions are highly imbalanced, since the database with the targeted classes are mostly imbalanced.

One drawback of these acoustic sensor is, that these sound information collected by a microphone may also contain speech information and thus raise privacy issues. A viable solution is to use surface vibrations instead of sound signals.

Surface acoustic sensors Pan [PWQ*15] built a person identification system that utilizes footstep induced structural vibration. The system can sense floor vibration caused by footstep without interrupting human activities. Gait analysis using the characteristics of individual footstep is then exploited to achieve an identification accuracy of 83 %. By further incorporating a confidence level, the accuracy rate can increase up to 96.5%. This is done by using only the most confident traces above certain threshold.

The signal to noise level of the received structural vibration signal is highly dependent on the sensor coupling to the ground and the surface materials. A sound coupling provides a higher signal-to-noise ratio. However it is also possible to increase the detection accuracy by performing more signal processing on the input stage. Since these acoustic events contain high frequency component, neglecting the low frequency components of the vibration signal further concentrates the signal energy to a smaller frequency bands and thus further improves the signal-to-noise ratio. Mirshekari [MPZN16] managed to improve the localization accuracy of indoor footstep signals in this way. They were able to achieve an average localization error of less than 21 cm, resulting in an improvement of 13 times compared to the use of the raw input data.

Alwan [ARK*06] proposed a work to detect the fall event by leveraging a seismic sensor to catch the distinctive vibration characteristic of a fall event. Falls are most common among elders and are one of the leading cause of death for elders. The authors worked to distinguish patterns from dropping objects close to the sensor and simulated fall events from a Rescue Randy up to 20 feet away from the sensor. The detection of a fall event is extracted from the models according to the vibration patterns, such as frequency, amplitude, duration, and succession.

Ultrasonic Sensors Ultrasonic sensors are active sensors, which actively transmit and receive signal to remotely perceive its environment. Ultrasonic spectrum starts from 20 kHz to 200 MHz, that is just above the human audible range. Ultrasonic sensing can be conducted in several classical forms. Acquiring distance information only, a pulsed sensor can be used to transmit high frequency pulsed signals and await for the reflected pulse bounced back off the measuring object. The operation frequency for most of the ultrasonic distance sensor are chosen to be 40 kHz. The time of flight, when the echo is registered by the ultrasonic receiver are correlated to the distance. The equation for calculating the object distance is thus $D = \frac{v_0 \cdot t}{2}$, where the speed of ultrasonic wave through the air is $v_0 = 340 \frac{m}{s}$ at a temperature of 20°C. Notice the 2 is the round-trip of the echo signal.

Acquiring motion information, such as the relative speed or moving direction, the Doppler measurement is required. To measure the quantity of Doppler broadening, a continuous signal of 20 kHz is emitted by the transmitter. The relative motion of a moving object is modulated above this carrier frequency. The amount of the Doppler in frequency shift can be calculated by using the Doppler equation, which then directly renders the information regarding speed and the sign is related to the direction of the relative movement.

Indoor activities, especially activities of daily living, such as *standing, sitting and falling*, and *Quantified-self* are the most popular use-cases for using ultrasonic sensors. Notably, for recognizing simple indoor activities, pulsed ultrasonic sensors are often used to measure distance towards the interacting object. Ghosh et al. [GSC*17, GCC*19] mounted 4 HC-SR04 sensors to cover a square of 70 cm x 70 cm with a LV-MaxSonar-EZ0 in the middle to reduce the dead zone. Relying on the distance profile, they used the support vector machine (SVM), k nearest neighbours (k-NN), and Decision Tree approaches to classify the targeted activities. The ac-

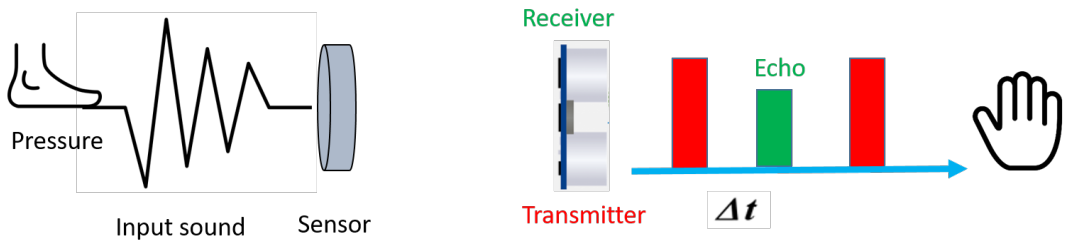


Figure 2.3.: On the right side, the principle of a surface acoustic wave is depicted. Each footstep causes the surface to vibrate. This vibration can be measured by a microphone or seismograph. On the left side, a pulsed ultrasonic signal is depicted. Range information is unambiguous within two subsequent pulses.

Sensor	Detection Range	Field of View	Operation Frequency	Noise Filter
HC-SR04	4cm - 400cm	15° - 30°	40 kHz	required
LV-MaxSonar-EZ0	15.2 cm - 256 cm	33°	20 kHz	required

Table 2.1.: Table lists commonly used ultrasonic sensors applied to build human activity recognition system in indoor spaces.

tivities contain primitive activities such as *sitting, standing* and *fall*. Using Hidden Markov Model (HMM), they later extended their work to recognize these events for a group of multiple person [GCP*18] and the transitions of these primary states. Patel [PPA18] targeted at a complete new set of activities of daily living including (*Nothing, Entered, Using Refrigerator, Used Refrigerator, Appeared near burner*, and *Using burner*) by applying Fusion of sensor networks consisting of Infrared Breakbeam Sensor, Ultrasonic sensor(HC-SR04) and Passive Infrared sensor(HC-SR501). The sensor specifications for the leveraged ultrasonic sensors are illustrated in Table 2.1. The operation frequency of the sensor, its field of view and the detection range are provided.

Physiological signals can likewise be detected by using a ultrasonic signal measuring the distance modulation of the chest movement during a respiration circle. Nandakumar [NGW15] developed a contact-free sleep apnea detector with an off-the-shelf smartphone. They transformed the phone to an active sonar system by emitting linearly frequency modulated sound signals (from 18 kHz - 20 kHz) and extracted range information from the reflected echo signal caused by the chest movement. Hand gesture recognition tasks using a smartphone device is further targeted by the project *Dolphin* [QHX*14] and *FingerIO* [NITG16]. Due to the limited detection range of a ultrasonic device, for close-range and fine-grained detection such as hand gesture and chest movement, a mobile application is more suitable than a fixed installation with a pulsed ultrasonic device.

Discussion As stated in previously cited works in Subsection 2.1.1, acoustic sensors, such as microphone, are mostly used to detect sound events, such as coughing, chair moving, door slam, transmitted through air. They are commonly used to infer sound-based events in private or public areas, such as a meeting room. Acoustic sound event is one of the most informative source besides natural speech to interpret a scene containing human beings and their interaction with the environment [SWvHG11, TN06]. These sensors do not require a solid coupling between the transmit medium and the sensor itself. However due to the nature of sound events, these sensors may raise privacy issues, since the general speech could be interpreted by the microphone.

Surface acoustic sensor measures the structural vibrations transmitted through solid materials. Since the production cost of these sensors are relatively low, they are often used to build distributed systems. It is power-efficient and its sparsity can further reduce the installation and computation costs. Applications built with this sensor type are mostly focusing on events causing vibrations on the ground surface, such as step signals [PWQ*15], object dropping or fall events [ARK*06]. These events form a primitive set of activities of daily living in a household. However, sensors leveraging the structural vibration require a solid coupling between the sensor and the solid material. If the load on the ground surface is changed, the vibration intensity and the pattern previously extracted will also be deformed. These effects often lead to drops in the detection performance and require sensor calibration.

Ultrasonic sensors overcome both disadvantages, by transmitting and receiving high frequency signals to unobtrusively perceive its environment. The operation frequency is above the audible range of a human being and thus the audible spectrum can be excluded for processing. Opposed to surface vibration signals, no coupling to the ground is necessary. Integrated into the environment, it can sense object up to 2 m with a pulsed sensor operates at 40 kHz. Relying on the distance profile, activities such as sitting, standing, and fall events can be recognized [GCP*18]. Operating in close range, it can detect fine-grained activities, such as hand gestures [QHX*14, NITG16] or even respiratory rate [NGW15].

The usage of these sensor categories in the domain HAR are three-folds,

1. sound events detection related to natural sounds from activities of daily living with microphones,
2. surface vibration detection due to step signals with surface acoustic sensors,
3. dynamic activity recognition with ultrasonic sensors.

Take-Home Massage One can notice that most works related to activities of daily living requires a network of this types of sensors. Due to the limited detection range of this sensor type, a full coverage of a room-scale requires multiple sensor fusion. Sound events, such as coughing, chair moving, or door slam can be detected by microphone arrays. Surface-bounded events, such as steps or falls are mostly measured by surface acoustic sensors. Fine-grained gestures or other delicate physiological signals require a close sensing range and high resolute senor system. For these applications, ultrasound sensors are preferred. An overview of the cited literature can be found in Table 2.2, where the previous works are introduced in terms of its application area, sensing device, processing algorithm, sensor behavioral, database and a concluding remark.

2.1.2. Electric

The strength of an electric field is related to the amount of charge produced by an electrified object. When a detection electrode is placed close to an electrified body, an electric charge proportional to the amplitude of the electric field is induced in the detection electrode. This physical effect is called electrostatic induction. The electric field can also be modified due to capacitive coupling with conductive materials or any other materials with a dielectric constant other than air. In the following, this subsection will introduce two main categories of this sensing technology: capacitive proximity sensing and electrostatic sensing with electric potential sensors. This subsection will later include an overall discussion of the technology and conclude with some final thoughts.

Capacitive Proximity Sensing Capacitive measurement is based on electric field proximity sensing relying on the fact that an electric field is perturbed by the existence of a nearby conductive object, such as part of a human body. Therefore, this technology is often applied for remote sensing in the field of HAR. Capacitive sensing principle can be further divided into three operation modes, ranging from *loading modes, shunt mode*, and

Work	Device	Area of Use	Algorithm	Database	Remarks
[SWvHG11]	Microphone	ADL	Self-organizing map (SOM)	Private	Sound events in office
[TN06]	Microphone	ADL	SVM, GMM clustering	RWCP, ShATR, Web	Sound events in public areas
[PWQ*15]	Geophone	Identification, Gait Analysis	SVM, Multi-class C-SVC	Private	Hardware design required (Amplification circuit)
[MPZN16]	Geophone	Indoor Localization	TDOA	Private	Sensor networks required
[GCP*18]	Ultrasound Device	ADL	HMM	Private	Echos, distance profiles, heterogeneous ultrasonic sensors
[NGW15]	Ultrasound (Mobilephone)	Physiological signal	Peak detection in time series	Private, clinically collected	Less power efficient, since continuous sensing
[QHX*14]	Ultrasound (Mobilephone)	In-air gesture recognition	Linear SVC	Private	Multiple mobile devices
[NITG16]	Ultrasound (Mobilephone)	Fine-grained finger tracking	Trajectory tracking	Private	Enhanced interactive Display, OFDM signals, 2D finger position by leveraging two microphones

Table 2.2.: Applications build on acoustic sensing.

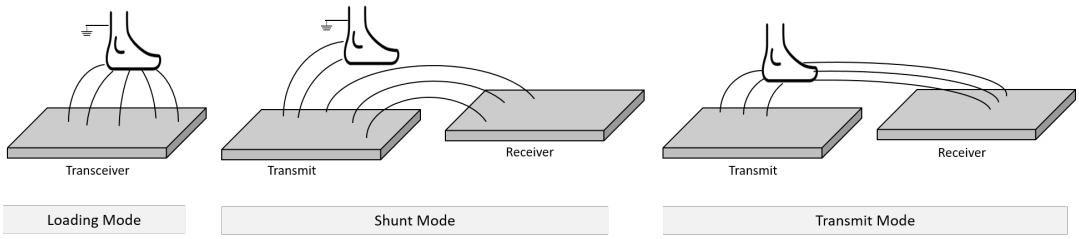


Figure 2.4.: Figure depicts the three operation modes of an active capacitive measurement [Smi96]. The electric field lines are depicted in black and are orthogonal to the surface.

transmit mode, according to Smith [Smi96]. In capacitive proximity sensing, the sensing category applies voltage to one side of the sensing electrode generating a constant electric field. The presence or motion of a conductive object close to the sensing electrode perturbs this electric field. The amount of the perturbation is directly correlated with the interactive item placed close by the sensing electrode. In Figure 2.4, the three operation modes of an active capacitive measurement are depicted. In *transmit mode*, the object acts as a transmitter and shortens the path of the electric field lines and amplifies the electric field. When the object is far from the receiver, the electric field weakens with $\frac{1}{r^2}$, since the object acts as a point source. Here r is the distance between the object and the receiver. While the distance decreases, the electric field weakens with $\frac{1}{r}$, as in this case the object acts as a parallel plane object to the receiver. In the *shunt mode*, electric field lines are partially occluded by the object and weakens the electric field strength. In the *loading mode*, one can measure the displacement current from a transmitter electrode to a grounded body part. This mode is often used to get the relative distance from the sensing platform to the object.

Nowadays, capacitive technology can be found in almost every smartphone, tablet or touchscreen display. It is affordable and can detect the presence of fingers, hands or body movement with high accuracy. The project **Touché** by Sato [SPH12] intended to enhance the touch interaction with capacitive sensing technique by leveraging the sweep frequency capacitive sensing technique. Conventional capacitive sensor operates at a certain frequency and can only detect touch interaction due to the amplitude modulation. By leveraging multiple frequencies, a more advanced profile can be built to include a variety of information, such as distinguishing between *not touching, touching, pinching*, and *grasping*.

Enhancing the touch modality, researchers design applications leveraging the proximity sensing ability of capacitive sensing. Proximity enables a more natural form of interaction compared to basic touch interactions. Braun [BFMW15] proposed a driver's seat enhanced with capacitive proximity sensing to detect a wide range of physiological parameters about the driver and his sitting postures for activity recognition in automobile applications. Identifying lying postures in bed, such as *supine, right lateral, prone, and left lateral* has been proposed by Lee [LHL*13] using the ECG signal of 12 capacitive coupled electrodes horizontally integrated into a bed cover. Rus [RGPK14] proposed similar lying posture recognition with mutual capacitance as sensor grid deployed under the mattress. These applications integrate the sensor electrodes into individual objects close to the sensing body.

Large-scale systems can also be built using capacitive sensing technique. Steinhage [STS*13] proposed a smart floor using capacitive sensing that can be embedded under any non-conductive surfaces such as carpet or stone. Multiple features, such as person identification, persons path or trajectories tracking and fall detection are developed for this application. These features are especially useful to elderly care facilities. Similar work, **TileTrack** by Valtonen [VMV09] applying transmit mode, to measure the capacitance between multiple floor

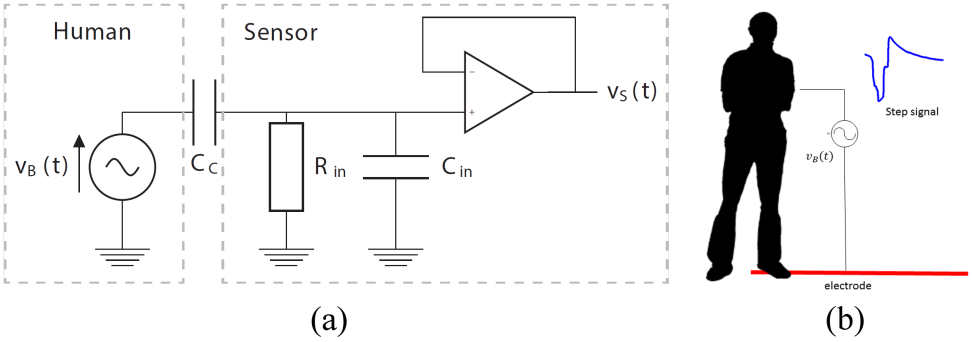


Figure 2.5.: Figure shows the working principle of an electric potential sensor on the left and a typical step signal induced by the displacement current from the body motion [FKvW*18].

tiles and the receiver electrode to perform indoor 2D localization. The system with an operation frequency of 10 Hz can localize a standing human with an accuracy of 15 cm and a walking person within an error range of 41 cm.

Applications with capacitive sensing introduced so far are commonly focusing on static or stationary measurement such as sitting or lying postures and thus more stationary information are provided. Dynamic nature of the whole-body interaction and other remote activity recognition is sparsely exploited. This is partly due to the physical principle of static field measurement, but also a lack in this research direction.

Electrostatic Sensor Electric potential sensor is an electrostatic sensor. Unlike capacitive sensing actively keeps a constant electric field to the sensing electrode, electrostatic sensor works with stationary electric charges. Electrostatic involves building up charge on the surface of objects due to contact with other surfaces. This charge induces an inverted charge on other opposite surface. Therefore electric potential sensors can be operated more power efficient due to the passive measurement of induced charges. However this induced charge is only noticeable, if the other surface has a high resistance to electrical flow and thus making the process of discharge remains long enough to be observed. Passive electric field measurement on the opposite is strongly dependent on the dynamic nature. The measurement solely uses body movement to generate body charges induced onto the sensing device. In case of electric field sensing, no constant electric field is applied on the sensing electrode. The sensing is merely due to the modulation of the existent ambient electric field caused by charge redistribution due to human motion. Thus, this sensing technology is strongly coupled to the ambient changes. The advantage of the electric potential sensors are light weight, large detection range, and high sensitivity. By using an ultrahigh impedance sensor at the input stage, even the smallest displacement current caused by the body motion can be reliably measured. The working principle of such an electric potential sensor is viewed in Figure 2.5. The modulation of body induced current is illustrated by using an oscillating voltage source v_B and it is changing over time. The displacement current from the body motion is coupled between the body's surface and the sensor's metal surface with a capacitance C_c, which is typically in the order of $0.1 - 10$ pF [FM11]. This weak capacitive coupling requires a very high input impedance to reliably detect the minor displacement current generated by the body movement. Normally it is in the order of $10^{12} - 10^{15}\Omega$, to keep the output voltage v_s stable.

Prance [PBRW*08] presented the ability of using an electric potential sensor to remotely detect physiological signals, such as the heart beat or respiration rate in a distance up to 40 cm from a seated subject. Rekimotor

[RW04] built an enhanced game-pad using electrostatic potential sensing to allow whole-body input interactions such as (jumping, landing, foot lifting and foot touch) besides the general key press input modality. However, since the sensing principle relies on body charge modulation via body motions, most applications are focused on wearable designs, such as the work by [PPPR16, MSU17, CGL*12].

Cohn [CMPT12] used a human body as an antenna for whole-body interaction in an indoor environment, by placing a miniature device on the body to collect the existent environmental "noises", such as AC power signal at 50 Hz or 60 Hz or other higher frequency signals from appliances and electronic devices. They leveraged the modulation of these electronic signals specific to different activities caused by the body motion. They are able to sense 12 activities with an accuracy of 93 %.

Remote and embedded installation for this sensing technology have been developed mainly for indoor localization purposes, such as in the works [GPDH*16, FKvW*18]. In the project **Platypus**, Grosse-Puppendahl [GPDH*16] showed that by installing four ceiling mounted electric potential sensors covering an area of 2 m x 2.5 m, they were able to track people in a nearly empty office room around 16 m^2 with a mean localization error of below 16 cm. They found out that the electric pattern for each step for different person are distinctive within a short time window. Thus making use of the pattern recognition with handcrafted features by integrating priors from domain expert knowledge and using some common features from literature regarding gait analysis, they were able to re-identify four users with an accuracy of 94 % and 30 users with an reduced accuracy of 75 %. Fu [FKvW*18] deployed the measuring electrode in a grid-wise layout under a non-conductive floor covering to perform indoor localization. With a sensor electrode spacing of 20 cm and an system operation frequency of 10 Hz only, they achieved a mean localization error below 12.7 cm by leveraging a weighted mean position estimation method. The sensing area covered an area of 240 cm x 360 cm in a simulated living laboratory environment. The author stated, that this sensing technique is strongly dependent on the foot-wear of the users. The strength of the induced charge is strongly dependent on various aspects, such as the clothing, weather condition and foot wear, which makes the sensing system extremely susceptible to environment noise.

Discussion According to the cited works in Subsection 2.1.2, capacitive proximity sensing is commonly used to sense direct interaction modality such as touch interactions. It can also be applied to detect conductive objects up to 15-50 cm and thus enabling other applications expand the touch interaction. The sensing technique is well suited for measuring stationary objects, such as postures or other stationary information in close range to the sensing electrode. Thus for close range activities and stable detection, the active capacitive technique is more preferable. Capacitive technology is widely used in touch screen technologies of the most current smart screen devices [Ors12], such as smartphones, tablets or touch screens. Besides the basic touch interaction, the most common usage of capacitive proximity sensing is in static posture detection, such as sitting postures [BFMW15], lying postures [RGPK14] or falling events [STS*13]. Large-scale installation is leveraged for indoor localization task [VMV09] or reasoned to build system performing recognition of activities of daily living [STS*13].

Technique of electrostatic sensing is used to better measuring the dynamic activities. In this case, the sensing is due to the surface charge generation caused by movement. The produced surface charge induces an inverted charge on the opposite surface that is measured by a sensor with a relatively high input impedance. This type of sensor is light-weight, easy to deploy and power efficient, since no active electric field is generated and only the existent ambient electric field is exploited. This kind of sensor is applied in various use-cases ranging from sensing of physiological signals [PBRW*08], to dynamic human activities [RW04], such as jumping, stepping or walking. Room-scale activity recognition [GPDH*16, FKvW*18] with this kind of sensor is also possible. Even with a relatively low system operating frequency of only 10 Hz, an accurate indoor positioning system is achievable. Build upon this trajectories, researchers can easily conduct other advanced researches such as gait analysis

or behavioural analysis of the inhabitants. Combined with a reasoning system, Kirchbuchner [SKvW*18] carried out predictions for early detection of dementia or other mental deceases.

The usage of this type of sensors in the domain of HAR are two-folds:

1. close-range postures and stationary action detection with proximity capacitive sensors,
2. passive, far-range dynamic activity detection with electrostatic sensor.

Take-Home Message Capacitive sensing technique is commonly used to detect stationary activities in close range, either direct touch or proximity up to 15 cm. Most common applications are finger touches, human postures or indoor localization. The resolution and detection range is directly related to the size, material and applied voltage on the sensing electrode. Capacitive sensor can produce ambiguous measurements. Placing a small object close by results in the same measurement as a large object placed at a distant distance. This issue should be considered during the design phase. However the signals are consistent, such that it provides reproducible signals for same object under same measuring condition.

Electrostatic measurement of the electric potential sensor is commonly used to detect dynamic changes, such as body movements. The detection range of up to 2 m relative to the hardware application is huge with respect to capacitive proximity measurement. However, the disadvantage of this sensing technique is that it is extremely susceptible to environment noise, which should be considered in the data processing stage. The signal patterns within a very short window is only reproducible, thus making it difficult to extract robust features directly on the signal pattern in time. The binary information of movement or non-movement can be leveraged to build precise indoor localization systems. Based upon the trajectories advanced applications can be researched. An overview of the cited literature can be found in Table 2.3, where the previous works are introduced in terms of its application area, sensing device, processing algorithm, sensor behavioral, database and a concluding remark.

2.1.3. Mechanical

Mechanical signal often indicates the force applied to a surface. The quantity of surface deformation is hence related to the impact of the interactive object. This can be expressed by the term $P = \frac{F}{A}$, where P is the pressure, F is the force applied in the normal direction to the surface and A represents the area of contact. The force induced deformation of the sensing surface, generating an electric signal, which is sampled by an analogue to digital converter to a quantitative measure. There have been many developments of pressure sensors in the past, which vary in terms of performance, technology, design and cost [SSSY17]. Its main application areas can be found in industrial monitoring, such as flow measurement or leakage detection [SSSY17]. In this subsection we will discuss two main categories of this sensing technology: resistive pressure sensing, and room-scaled pressure sensing with piezoelectric or fiber optical sensors. This subsection will discuss these two categories and later include an overall discussion of the technology and a final conclusion.

Resisitive Pressure Sensing Applications for HAR has been proposed in [XHA*13, SCZ*14, CSZ*16]. Xu et al. designed a *eCushion* to detect sitting postures. They used the resistive technology to measure the surface deformation by integrating fiber-based yarn which is coated with piezoelectric polymer [RXS10]. The initial resistance of an unstressed surface is relatively high. With force applied to the textile, the intra-fiber distance will be squeezed and thus causing the resistance to drop. By performing signal matching with dynamic time warping method, they achieved an overall recognition accuracy of 85.9 % for 7 sitting postures.

For Quantified-self applications, Sundholm [SCZ*14] developed a flexible sport mat equipped with a thin layer of conductive polymer fiber sheet consists of resistive pressure sensor matrix. The conductive sheet is positioned

Work	Device	Area of Use	Algorithm	Database	Remarks
[SPH12]	Cap	Smart Objects	Sweep Frequency Capacitive	Private	Exploratory Study
[BFMW15]	Cap	Sitting postures, physiological signal	Model-based approaches	Private	Sparse sensor setup
[LHL*13]	Cap	ADL	LDA, SVM, ANN	Private, lab. setup	12 capacitive coupled electrodes horizontally placed on the bed
[RGPK14]	Cap	Lying postures	SVM	Private, lab. setup	Sensor grid integrated under the mattress
[STS*13]	Cap	Indoor Localization	Kalman Tracking	Private	Sensors are modular
[VMV09]	Cap	Indoor Localization	Threshold in time series	Private, constrained environm.	Resolution is proportional to sensor size
[PBRW*08]	EPS	Physiological signal	Peak detection in time series	Private, medical center	Remote measure of heart beat
[RW04]	EPS	Whole-body Gesture sensing	-	Private, constrained lab. setup	Enhancing interaction for entertainment, Exploratory study
[CMPT12]	EPS	Presence detection	Threshold	Private	Leverage the ambient electric power lines
[FKvW*18]	EPS	Indoor Positioning, Identification	SVM	Private, office room environm.	Ceiling mounted
[GPDH*16]	EPS	Indoor Positioning	2D Positions	Private, living lab. environm.	Smart Floor installation

Table 2.3.: Applications build on electric field sensing.

between 80 parallel stripes of conductive foil on each side (horizontal and vertical), resulting in a 80 cm × 80 cm sensor mat. The volume resistance of the fiber sheet changes locally, when the material is pressed. As output, a 80 × 80 pixel frame of the applied pressure can be sampled at 40 Hz. They recorded 10 exercises of 7 users, each exercise repeated 10 times over 2 different sessions per subject. These exercises include workouts such as *push-up, quadruped, abdominal crunch, bridge, etc*, and additional weight training such as *chest press with dumbbell* and *biceps curl with dumbbell*. An overall classification accuracy of 88.7 % for the person dependent and 82.5 % for the person independent case can be achieved with a k-NN classifier. Template matching with dynamic time warping method is applied to count the repetitions. An average counting accuracy of 89.9 % across different subjects is achieved.

Room-scale Integrated Pressure Sensing Other installed and embedded applications are focused on indoor positioning or detection of activities of daily living [AJLS97]. Integrating pressure sensors into furniture and floors in home environment, Lim [JHJP08] was able to recognize daily activities such as *meal, sleep, exertion, go-out*, and *rest* base on the object usage information. If anomalies in a healthy daily living style are detected, a warning sign can be provided to care-givers or doctors without intrusion.

Similarly the **GravitySpace** [BHH*13] is an instrumented space used to track the user's location and their poses relying on the physical imprints of the human force impact left on the sensing ground. Integrated with other modalities such as marker-based motion capture systems, audio-sensing equipment and video-sensing technology, Srinivasan [SBQK05] provided the pressure information as an additional input modality to enhance the application for interactive media usages.

Finally, the pressure can be measured not only with resistive technology, but also with fiber optics, as demonstrated in multiple works [FMB*16, NSN*10, WGLK16]. Feng [FMB*16] used floor pressure imaging for posture-based fall detection with fiber optic sensor grid-layout embedded under the floor space. People identification with gait analysis has been targeted in the work by Qian [QZK08]. Using a large area, high resolution, pressure sensing floor, they were able to provide 3D information of each footstep (containing the quantity of force and the 2D positional information). Applying the fisher linear discriminant classifier on the collected patterns from these 3D data points over time for each participant, they obtained an average recognition rate of 94 % and a false alarm rate of 3 % by using pair-wise footstep data from 10 subjects.

Discussion In accordance to previously discussed works in Subsection 2.1.3, we identified that pressure sensor arrays integrated into flexible textiles can be used in the applications for posture sensing or activity sensing. Build upon sitting posture recognition, researchers retrieve high-level contexts leveraging these primary information. Mota [MP03] tried to associate these naturally occurring postures and corresponding effective states relate to a child's interest level while performing a learning task on a computer. Features were extracted by leveraging a mixture of 4 Gaussians to express the force distribution on the back of a chair. A 3-layer feed forward network was used to train the classifier for nine postures and an overall accuracy of 87.6 % was achieved for testing on new subjects excluded from the training set. A set of independent Hidden Markov Models was used to link to three categories related to a child's level of interest. An overall performance of 82.3 % with posture sequences from known subjects and 76.5 % with unknown subjects were realized.

Textiles based prototypes are flexible and easy to transport, however, they suffered from the problem of maintainability. Since the force is directly applied to the sensing surface, a flexible surface could be slightly deformed every time, it is used. Cheng [CSZ*16] also noted that every time the Smart-Surface is installed and used, it is twisted slightly differently, which leads to a different default pressure distribution asserted by its own weight and folding. Further problems of textile sensors noted by Almassri [AWA*15] such as non-linearity, drifting and hysteresis could also influence the generality of the developed model for the target application.

Pressure sensors embedded under any floor covering or integrated into furniture as part of a distributed sensor networks can provide large scale sensing in contrast to portable systems. They can be used to sense room-scaled indoor information such as location or other activities of daily living. Integrated into furniture or objects, these objects can provide usage information to be accessed for smart home applications. Depending on the footstep force profiles, Orr [OA00] proposed a floor system to identify users in their everyday living and working environments. Creating user footstep models using footstep profile features allows them to achieve a recognition accuracy of 93 %. They've further shown, that the effect of footwear is negligible on recognition accuracy, in contrast to other sensor types, such as electrostatic sensing technique. Thus pressure sensors installed as a floor-based system enables a more robust and natural identification of users.

The usage of this sensor category in the domain of HAR are two-folds:

1. close-range posture, or action detection with flexible, resisitive textiles ,
2. room-scale sensing with either distributed pressure sensor networks or installed floor-based applications.

Take-Home Messages Mechanical sensor works with pressure profiles caused by impact. Hence direct interaction is required. It is similar to active capacitive measurement by leveraging stationary force impact. Therefore, mechanical measurement is ideally used to measure postures or stationary activities. However, the proximity sensing would provide more information, including close range interaction as an additional input modality complementing the direct touch. Compared to passive electric field measurement, the foot-wear is negligible on the recognition accuracy for pressure sensing applications [OA00]. Thus this type of sensing technique is more error-resistant to the surrounding environmental noise, but bears the inherent problem of easier deformation. An overview of the cited literature can be found in Table 2.4, where the previous works are introduced in terms of its application area, sensing device, processing algorithm, sensor behavioral, database and a concluding remark.

2.1.4. Optical

Optical sensors can quantify the intensity of light. Optical spectra cover a wide frequency range, from ultraviolet (280 nm - 360 nm) to visible (380 nm - 750 nm) to infrared (800 nm - 1000 nm). Invisible infrared light spectrum can be detected by infrared sensors, while visible light can be measured by the charge-coupled device (CCD) of a standard camera. In this subsection, we concentrate on the imaging ability of these optical sensing devices with the focus on HAR. We discuss three main categories of this sensing technology: visible imaging, depth imaging, and thermal infrared imaging. This subsection will later include an overall discussion of the technology and a final conclusion.

Visible Imaging Vision-based HAR is probably one of the most well researched area in the field of computer vision, for enhancing the human machine interaction interface. Vision input compared to time series from sensor data provides more contextual information. From outdoor security applications [SBTM08], integrated with virtual reality techniques for entertainment purposes [SH17], monitoring and analysing sport activities [LL06, LWH03, LLO04], to medical applications [KJvdS17], the demand on mature computer vision algorithms is growing.

Starting from segmentation [HGDG17b] and recognition of human poses [CHS*18], towards continuous HAR [AA01], the full chain has been well studied. The most difficult part is to find feature representations in images to help developing robust human action modeling and thus improving the ability of algorithm to classify the correct activities. Unlike 2D image space, challenges in video sequence classification may include different appearances, shapes and poses in video frames over time and problems of occlusion from subsequent frames. From

Work	Device	Area of Use	Algorithm	Remarks
[RXS10]	Piezoelectric polymer	Sitting postures	DTW	eCushion: challenge in form stability
[MP03]	Pressure Matrices	Sitting postures	MoG [Das99] + ANN	relate naturally occurring postures to child's interest level
[SCZ*14]	conductive polymer fiber	Exercise recognition	k-NN	Flexible sheet induces the problem of deformation
[AJLS97]	Weight sensitive floor	Indoor Positioning, Footstep signatures	HMM	No EMC problems
[JHJP08]	Pressure sensors	ADL	Rule-based, attached to binary sensor states	Integrated into furniture and floors in a home environment
[BHH*13]	Pressure sensing floor + touch sensitive furniture	Identify users, furniture and poses	Clustering for poses, Matching database for identification	Updates in real-time and runs a physics engine to model the room above the surface
[SBQK05]	Pressure sensing floor	Dance movement	Pressure pattern from a footstep	Multimodal sensing of a subject, including motion capture, audio and video sensing systems, high density of sensors required
[FMB*16]	Fiber Optics	Fall detection in smart home application	Pressure image and rule-based classification	false positives due to missing dynamic links
[NSN*10]	Embedded hetero-core fiber optic nerve sensors	ADL in bathroom	Pressure mapping	Activities in bathroom, prone to temperature fluctuations and electromagnetic interference
[QZK08]	High resolution pressure sensing floor	Gait Analysis, People Identification	multi-classes FLD	Strict assumption that features from different persons are linearly separable

Table 2.4.: Applications build on mechanical sensing.

carefully handcrafted feature representations with induced prior knowledge [DDS16,NSY15,WQT15,ZHX*07], to the earlier stage of the deep learning era, a lot of efforts were made on developing robust models and generalized feature representations for accurate activity classification. Convolutional neural networks (CNN) like AlexNet [KSH12], showed its superior ability to automatically extract useful feature representations from the underlying data structures. Other generative models, such as sparse autoencoders [N*11], and generative adversarial networks [RMC15], are representatives of methods able to automatically learn the embedding representations of data.

Tran [TBF*15] studied a deep learning architecture for video action classification by extending a conventional 2D-CNN with a third convolution direction over time. The structure is called *C3D*. Their work showed that this type of network is especially designed to extract features that model appearances and motion simultaneously. Input to the network is video clips of the dimension *l*x*w*x*h*, where *l* represents the number of frames per clip, *w*x*h* stands for the width and height of a frame and the output is the class probabilities of each activities. The network consists of several consecutive convolution and pooling layers to extract the high level appearance features and expand the field of view of the locally connected convolution features. However, it is to note that the first pooling layer only reduces the spatial dimension, but not the time dimension in order to preserve the temporal information further in the network. The performance was evaluated on three public available video databases such as the **Sports-1M** [KTS*14], **UCF101** [SZS12] and **YUPENN** [DLDW12].

Another common design for video classification is the Two-Stream approach by Diba [DPG16]. They showcased a similar model using two streams of 3D CNN. Such architectures are intended to solve the problem of insufficient training data as well as noise introduced by different view points, perspectives and variation in motions. The first branch, referred as the appearance stream, implements the regular *C3D* network, while the second one, referred as the motion estimation stream, uses optical flow as input. The features from the two streams are concatenated and feed to a softmax layer to infer the probability distribution of the classes. While testing on the UFC101 dataset [SZS12], the two stream model outperformed the *C3D* network by 5 % with a 20 % decrease in processed frames per second. It confirmed the assumption that using optical flow helps the network recognize motion and complements the appearance and spatio-temporal features learned by the standard *C3D*, however at the cost of increased computational performance.

Depth imaging The skeleton offers a more compact representation of the human body and enables simplified segmentation task and estimation of pose. Commercial products such as Microsoft Kinect makes visible images with depth information affordable. These devices can be used to capture human motions and provide the 3D coordinates (x, y, z) and the angle of the joints of the skeleton. The development of these skeletons over time in successive frames can be used to classify human activities of subjects within the measuring area. Compared to 2D images, the depth information facilitates the extraction of fore- and background.

Mostly, Microsoft Kinect is used to provide a depth channel in addition to visible channels. Official algorithm are provided to determine the skeletons and joint positions as features for various activity recognition tasks. Mettel [MASB19] introduced a fall detection service using a single depth camera installed on the ceiling. Combining static and dynamic methods, a fall detection service was achieved by using a Microsoft Kinect. A random sample consensus (RANSAC) method was used to estimate the ground plane. Static detection investigates whether the person is lying on the floor by tracking posture using skeleton joint data. Dynamic detection checks whether a person is previously falling to the ground by threshold the speed of the previous joint motion towards the ground plane. However, by placing only one single depth sensor in room, the sensing area is restricted thus leading to performance degradation, when the skeleton tracking is occluded by obstacles within the sensing area. Author proposed to use fusion of multiple installations to reduce false positives.

Cippitelli [CGGS16] proposed an activity recognition framework to exploit skeleton data extracted by RGB-depth camera for recognizing activities relevant for assisted living. Their promoted use-case is to provide help to monitor aged people in home environments. Their main contribution was able to automatically extract key poses without a learning algorithm. The key poses were extracted using a clustering algorithm to assign each human posture to the most important posture for certain activity. The key poses were then concatenated to build a feature vector that is used for the multiclass SVM to perform activity classification. The proposed algorithm is evaluated on five public available databases (**KARD** [GRM14], **CAD-60** [SPSS12], **UTKinect** [XCA12], **Florence3D** [SVB*13], and **MSR Action3D** [LZL10]) and showed promising results especially on a subset of basic activities designated from ambient assisted living scenarios.

GymCam [KAY*18] is a camera installation in a unconstrained environment, such as a university gym, which are then able to unobtrusively and simultaneously recognize, track and count fitness exercises performed by multiple persons. The promoted use-case is for Quantified-self applications. It involves several computer vision tasks such as correctly segmenting exercises from other activities, recognizing and tracking users performing the exercise by following the trajectories of the interest points and counting the number of repetitions. Based on motion trajectories from key-points tracking using dense Optical Flow method, they were able to classify different activities from these features extracted by these motion trajectories. The repetition counting was performed by template matching with an average trajectory of each exercise.

Thermal imaging Images from visible light spectrum, such as visible images, may face a problem in object segmentation, if the appearances of the human subject, e.g. the color of the clothing is indistinguishable from the background. Thermal infrared imaging is resistant to this effect and can provide complementary advantages. Thermal cameras are passive sensors to measure infrared radiations emitted by any warm objects. Therefore, human motion can be easily detected from the background regardless of lighting conditions and appearance changes [HB05].

To use computer vision in pervasive healthcare is not new. Camera system installed in a living environment to detect activities of daily living is introduced in the work [UNH08, WGH07a, DT01]. Person identification can be realized not only with biometric trait such as face images, but may also use soft biometric traits, such as gait pattern [GSD*13] or postures. To reduce the privacy concern regarding using cameras in domestic environments, low resolution thermal imaging method are applied to achieve the detection of activities of daily living without revealing a wide range of private information. Shelke [SA19] used two low-resolution (4x16) and contact-free thermal imaging sensors (MLX90621) to classify four different activities such as *stand, sit on chair, sit on ground*, and *lay on ground*. For static activities, such as sitting on a chair or standing still, frame-wise classification can be applied using conventional multiclass classifiers. Dynamic changes can be observed via shape changing effect from consecutive frames due to motion relative to the sensor according to lens projection equation. The shape can be detected by using connected component labeling approach [DSB99] to group the corresponding pixels. The disadvantage of using the MLX90621 thermal sensor is its limited field of view (FOV). It has a 120° horizontal FOV, but only 25° vertical FOV. Therefore, a careful arrangement of sensor placement is needed to achieve good performance.

Hevesi [HWP*14] leveraged a cheap (30USD), small, low power sensor array of 8x8 thermal sensors to unobtrusively and remotely detect a wide range of activities of daily living. The system can track people within the accuracy range below 1 m and detect the usage of electric appliances, such as toaster, water cooker or egg cooker. Basic activities, such as opening a refrigerator, the oven or taking a shower can also be detected. Due to the sparse sensor resolution by 8x8 pixels, the authors claimed that the system can be installed in the bathroom to recognize bathroom activities without invading privacy.

Kawashima [KKI*17] proposed a Deep Learning-based approach for action recognition method with an extremely low-resolution thermal image sequence. The hardware used is a grid of 16x16 far-infrared sensor array (Thermal sensor D6T-1616L by OMRON Corp.) mounted on the ceiling (around 220 cm above the floor) of a room. They focused on recognizing daily activities, such as walking, sitting down, standing up etc. and abnormal activities (e.g. falling down). The authors combined feature extraction method with shallow CNN structure (consisting of only 3 layers), combined with a sequence layer using long short term memory (LSTM) for extracting spatio-temporal representation. With a frame rate of 10 fps, the overall accuracy for the targeted activity classes were 85.75 %. Data collection consists of sequences from day and night times. The superiority against visible light is that the night vision for thermal imaging can make a "falling down" action in the dark visible in contrast to a total black visual input in visible light spectrum.

Discussion In accordance with the cited works in Subsection 2.1.4, camera systems provide richer information compared to other non optical sensors accompanied with the cost of more computation efforts. Recent advances made in computer vision domains ignite more interests in this field. Especially, faster progress was made in object detection and localization with algorithms such as YOLO [RDGF16] to faster YOLO [SCLW17], and Fast R-CNN [Gir15] to Faster R-CNN [RHGS15]. The tendency is to work on faster algorithms, which can be embedded on hardware with limited resources. The development from semantic segmentation with Mask R-CNN [HGDG17a] and Eye-MMS [BDKK19] to instance segmentation with DeepMask [PCD15] also allows a more precise information retrieval for separating instances from the same class. Video sequence processing with C3D network or attention network for sequence input [WJQ*17] make activity recognition in complex scene possible. Despite the advanced algorithms, camera systems still face challenges such as occlusion, change of appearance and prone to illumination changes, which are only partly resolved.

To reduce the negative effect of illumination changes, additional channel of depth can be integrated. The information of depth can be used to resolve the ambiguity in two dimensional image space. Commercial products from Microsoft and Intel make depth camera accessible for researchers to conduct experiments in the field of computer vision with depth channel. Microsoft Kinect automatically comes with the joint positions of the skeleton model. The skeleton representation is more sparse and compact, thus enabling more efficient processing on embedded hardware entities. Skeleton-based processing is commonly applied for human action recognition. Composite of extracted handcrafted features and well-designed classifiers, human skeleton can be used to extract spatial structure and temporal dynamics specific from human actions. Lately, research interests shift to consider end-to-end learning to avoid handcrafted features and model construction with prior knowledge. Du [DWW15] proposed an approach using hierarchical recurrent neural network to learn representations of skeleton poses hierarchically fused from sub-nets to automatically form action models fitted for the separate action classes. Skeleton-based approaches for HAR to build assisting system for elderly monitoring was introduced in [MASB19, CGGS16]. Activities of daily living, such as sitting, standing, walking, and falling are the most often targeted classes.

Thermal infrared imaging is another sensing form operating with near to far infrared light spectra. The operating wavelengths enable the system to observe radiations emitted by objects with a temperature above zero. Therefore facilitates the segmentation process from human object to background. Night vision capability of infrared sensors even enables action recognition in the dark opposed to image data from visible light spectrum. It also enables the reconstruction of visible-like images from thermal captures [DBM*19]. Infrared sensor arrays used in the cited works are mostly sparse and thus can be applied to reduce the resolution to protect users privacy. Sensor array of 4x16, 8x8 or 16x16 pixels are typically used. These installations are often applied in home environments to build systems for tracking and evaluating activities of daily living.

The usage of these sensor categories in the domain HAR are three-folds,

1. camera-based action recognition in public areas,

2. depth-based action recognition and tracking on embedded hardware platforms,
3. low resolution thermal infrared imaging in home environments to build ambient assisted living systems.

Take-Home Message Action recognition in computer vision can be performed on images, videos, or life streams. Each of the target domain bears its own challenges. Image covers only one instance in time and thus context can be missing if the decision relies only on one single image. Action recognition in video requires more complex network architecture to integrated the time component. Real-time assessment of human activities can enable robots to operate intelligently in interaction with humans. Part of these challenges have been already solved by the modern deep learning methods. By using 3D network structures or sequence modelling methods, the aspect of time is considered. Knowledge distillation [MM17] or network pruning [HS93] can decrease the model capacity and make real-time assessment possible.

Despite the rapid development in computer vision, one of the biggest drawbacks of camera based solutions is the low user acceptance in private sectors, as cameras typically raise concerns about privacy [Lan02]. Therefore, either using depth channel or using thermal imaging can help resolve some of the mentioned challenges for visible spectral input. An overview of the cited literature can be found in Table 2.5, where the previous works are introduced in terms of its application area, sensing device, processing algorithm, sensor behavioral, database and a concluding remark.

2.1.5. Radiation

Radiation, in the form of electromagnetic waves, works with high frequency electric field modulations. Common custom radar in the automotive domain operates at a typical frequency of 24 GHz [RHR10] and 76 GHz [FRL05]. On the other hand, according to WiFi standard 802.11n [SNC*08], domestic WIFI frequency bands operate at 5 GHz for close range and 2.4 GHz for far range. The operating frequency of 2.4 GHz grants for better penetration through solid objects and thus provides a wide coverage of WIFI signals. In the following, this subsection will introduce two main categories of electromagnetic sensors: radar sensors and WiFi sensors. This subsection will later include an overall discussion of the technology and conclude with some final thoughts.

Sensor devices generating a high frequent electromagnetic field, such as a radar, can operate in two different modes, in continuous wave (CW) mode and frequency modulated continuous wave (FMCW) mode. In the CW mode, only relative speed toward the receiver can be measured, while the FMCW can also provide distance information with the time beacon information encoded in the start frequency. In Figure 2.6, the two operation modes of radar is visualized. For continuous wave radar depicted on the left, if the transceiver and the distant object are both stationary, the received signal is not modulated. If the distant object is moving with a speed of v relative to the receiver, then a positive or negative Doppler shift can be measured for an approaching or departing object. Since there is no timing information available, only the relative speed represented by a Doppler profile can be extracted from the continuous signal. For frequency modulated continuous wave case depicted on the right, using the time shift of the received signal with respect to the transmitted signal, a distance profile can be generated in addition to the speed information.

WiFi sensing also depends on similar sensing protocols. However, it can further access the channel state information to infer HAR. Channel state information (CSI) describes the channel property between the transmitter and the receiver. Radio signal from the transmitter can travel directly to the receiver (LOS), but may also be scattered by objects or reflected by walls and ceiling before reaching the receiver. CSI can be represented by the channel transmission matrix, describing these different effects, such as fading, scattering, and multi-path fading, by the physical environment between transmitter and receiver. Common WiFi systems uses Orthogonal Frequency-Division Multiplexing (OFDM) [NP00] to divide the wide spectrum band into around 30

Work	Area of Use	Algorithm	Database	Remarks
[SBTM08]	Crowd behavior recognition	KLT tracking, crowd motion vector and clustering	Private	same type of event may be modelled slightly different from one scenario to another and many models are to be defined.
[GSD*13]	ADL, Gait Analysis	Model-based approach	KTH, Weizmann	Model-based approach is more constrained and less variable
[UNH08]	ADL, multi targets detection	Blob detection using Optical flow, Color-based tracking	Private, Smart Laboratory setup	Vision as an additional input modality to enhance the ability of existing no-vision sensors
[WGH07a]	Fall detection, localization	Threshold on aspect ratio, mapping image coordinates to world coordinates	Private, staged living room condition	not prone to false positives due to the missing dynamic links
[DT01]	Tracking of interacting and occluded human motion	Bayesian belief network for multi cameras fusion, Kalman filter	Private, simulated home environment	cannot handle self-conclusion
[SA19]	ADL	LR, NB, SVM, DT, RF, ANN	Private	Restricted field of view for thermal imaging sensors
[KKI*17]	ADL	CNN+LSTM	private, day and night vision of thermal inputs	16x16 grid resolution to preserve privacy
[KAY*18]	Gym exercises recognition and tracking	Optical flow, feature extraction of motion trajectories, MLP neural networks for classification	Private, Carnegie Mellon University's varsity gym	Limitations to handle intra-class variability
[NSY15]	Activity recognition	CNN + attention	NTU RGB+D, SBU Kinect Interaction dataset UFC 101, Kinects dataset	can handle noisy poses
[WQT15]	Activity recognition	Two-stream network	HMDB51, UFC101	Combine handcrafted features with deep learnt features
[ZHX*07]	Event detection in soccer broadcasts	Tracking with particle filters, SVM	FIFA World Cup 2006	Highlights and tactic detection, game analysis on broadcast videos

Table 2.5.: Applications build on optical sensing.

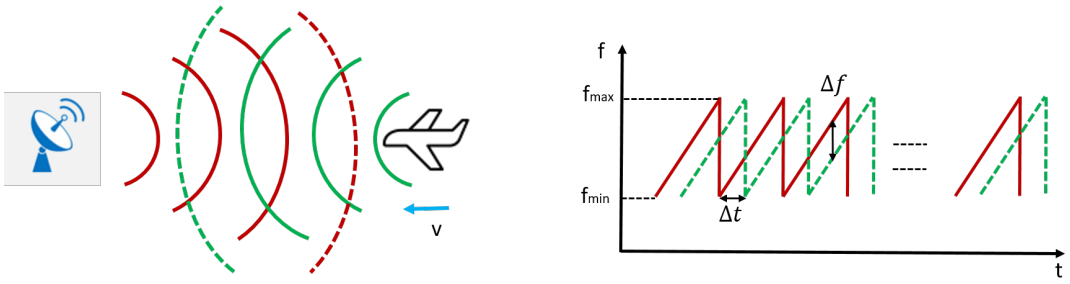


Figure 2.6.: Figure shows the two operation modes of a radar. On the left side, if both transmitter and object are stationary, then no Doppler shift can be measured. In case of a moving object with a speed v, a positive or a negative Doppler broadening will be measured relative to the motion direction to the transmitter. On the right side, the **FMCW** mode is depicted. Based on the time shift, the distance of the object towards the transmitter can be calculated using the time of flight. The transmitter signal is shown in red, while the receiver signal is depicted in green.

non-overlapping subcarriers. In this case, CSI contains complex values, which represents the channel properties of each subcarrier. Take a WiFi channel in the 2.4 GHz band with multiple inputs and multiple outputs (MIMO) mode, containing 3 Transmitter and 3 Receiver antennas, the CSI Tool can capture 30 OFDM subcarriers, resulting in 3x3x30 CSI data points in each received packet for processing [LCSL18] at each time instance. By collecting these CSI data points over time, we can build a CSI profile used to capture the changes in the physical environment. A moving object such as a human being in the receiving path can affect the channel response and be measured on the receiver side.

Radar sensors We start introducing the radar sensing, which has the advantages of insensitivity to environment conditions and robustness in different weather conditions. This makes radar applications in HAR a suitable candidate. It can transmit signal through walls, thus no direct line of sight is needed compared to vision-based systems. Using millimeter waves, the resolution is so fine that it can even detect the smallest finger movement in the order of sub-millimeter. Motion sensing with **Soli** [LGK*16], a tiny radar chip to detect and recognize hand gestures developed by Google has now been commercialized and integrated into Google's new smartphone *Pixel 4* [Die]. Rahman [RLBL*18] proposed yet another contact-free measurement of respiration rate by leveraging the phase shift in Doppler radar signal caused by the chest movement and allow person identification according to the subtle body kinematics of six individuals. A 2.4 GHz quadrature system is used to reduce the DC offset for more amplification and thus increasing the dynamic range of detection.

Seifert [SAZ19] used radar-based applications to perform unobtrusive person identification using In-home gait analysis. A K-band radar is used to collect data from four test subjects. K-band operates in the frequency range of 18–26,5 GHz, the radar used in their work operates at 24 GHz. In their proposed work, different walking styles were further clustered into five different gait classes including *normal, pathological* and *assisted walks*. By leveraging the radar micro-Doppler signatures, an average identification accuracy of 93.8 % was achieved across the classes and a classification rate of 98.5 % was achieved for a single gait class. A performance drop to 80 % accuracy was expected for unknown individuals. Features from both the spectrogram and cadence velocity diagram were extracted and used as inductive biases to the model. A simple classifier using nearest neighbour (NN) approach was applied to the handcrafted features condensed by the principle component analysis technique (PCA).

Liu [LPRS12] leveraged a dual Doppler radar system for fall detection operating at 5.8 GHz covering a detection range of 6 m. They leveraged the Mel-frequency cepstral coefficients (MFCC) [L*00] to extract features from the Doppler signatures caused by different activities. The decision of fall/non-fall detection was then determined by the fusion of multiple trained classifiers output.

Deep learning technique has also found its way to radar signal processing as in computer vision applications. Most of these methods were directly applied on time-frequency spectrum (spectrogram). Similar to computer vision tasks, where CNN is applied on images to extract features for object recognition, CNN can analogously be used on spectrum images to extract spectral patterns resulting from specific activities. Kim [KM15] proposed a deep convolutional neural network architecture for human detection and activity classification with Doppler radar operating at 7.25 GHz for outdoor and 2.4 GHz for indoor activity recognition with direct line of sight. This network jointly learn the feature representations and classification in one single network using the raw Doppler spectrum. Activity classes included are *running, walking, walking while holding a stick, crawling, boxing while moving forward, boxing while stand in place*, and *sitting still*.

Similar to time series for natural language processing, recurrent neural networks (RNN) can support the decision making stage of activity classification by considering the time aspect of the signal progress. However for radar images, a 2D-CNN layer is often applied prior to the RNN layer in order to extract robust features from the time-frequency spectrogram. The follow up work of using **Soli**, a customized, miniaturized radar chip to resolve sub-millimeter gesture motions, showed such a network structure in [WSL*16]. Their network consisted of two stages including the representation learning stage by using a CNN network, followed by the dynamic sequence modelling stage of using a long short term memory (LSTM) network prior to the classification stage with a Softmax layer. They achieved a per frame accuracy of 79 % and a per sequence accuracy of 88 % on a set of 11 hand gestures across 10 different users.

Ultra-wide band (UWB) is a radio technology that is used at short-range, high-bandwidth communications. It has been widely used in radar imaging domain. Compared to CW radars, it exceeds in terms of range resolution. Compared to FMCW radars, the UWB transmission is able to send very short pulses mitigating the multi-path inference problem. UWB operates commonly in the frequency spectrum of 3.1 GHz to 10.6 GHz, a broad frequency bandwidth of more than 500 MHz and a very short pulse duration of (<1 ns) [SF17]. This property makes the signal hard to detect and thus it is immune from detection, jamming, and interference. Lai [LN05] leveraged a UWB random noise radar to characterize human activities and through-wall imaging. So-far, the use-cases for radar imaging with UWB radars are mostly considered for military purposes or serve for law-enforcement. They can be used in the search and rescue operations. Ding [DZG*18] conducted a thorough investigation on a large number of motion types with an UBW radar system. They cluster different motions into two main categories of motion, including in situ motions and non-in situ motions. They leveraged physical empirical features for classifying in situ motions, such as standing, bowing, squatting etc. and PCA-based feature extractions for inferring non-in situ motions, such as walking, jogging, jumping forward and falling forward. They reported a final classification accuracy of up to 94.4 % and 95.3 % for in situ motions and non-in situ motions, respectively. They claimed that their proposed method could be also used in smart homes and senior care domains.

Radar is good for dynamic activity recognition, because of its robustness and its high resolution, as they operate at several gigahertz range, but it comes with the price of high power consumption and complex hardware design. WiFi devices are much more power efficient compared to radar sensors, if one can accept the comparable lower resolution.

WIFI sensors Most radar comes with a high specialization and integration between hardware and software packages. In order to fulfill certain task specification, a separation between the software layer and hardware layer are often needed. This makes embedded radar packages difficult to be specialized for a broad range of

applications in the HAR domain. Therefore, researcher tried to find a replacement which has similar physical behaviours, but are easier to modify and access. Researchers state that the channel state information (CSI) from a WiFi signal can be leveraged to passive and unobtrusively monitor the presence or motion of a human being. Popular application of using wireless devices for indoor localization using WiFi fingerprint is quite common, such as introduced in [TKY*16]. When a person comes in the way between a WiFi transmitter and receiver, it changes the received signal strength (RSS) transmitted to the receiver. This modulated RSS profile can be used to extract useful information with respect to activity classification.

WiGest [AYH15] is a ubiquitous wifi-based gesture recognition system to sense in-air hand gestures by leveraging the modulation in WiFi signal strength around a mobile device, such as a consumer smartphone. Based on three basic primitives, such as *approaching, removing*, and *holding above the device*, they were able to composite high level gestures without training for gesture recognition. With only one Access Point, they were able to detect the basic gestures with an accuracy of 87.5 %. To further include three Access Points, they were able to increase the accuracy to 96 %. Adding preambles as the start of a intended gesture, additionally improved the recognition accuracy and reduced the interference from multi-user scenario.

Accessing only the CSI of a WiFi signal, Zeng [ZWX*19] built an application to monitor human respiration even when the target is far away from the WiFi transceiver pair. Common WiFi based application needs the object to be close to the transceiver, because the attenuation of radio frequency (RF) signal operating at 2.4 GHz is around 6 dB for a solid wood door with 1.75 inches and almost 9 dB for an interior hollow wall with a depth of 6 inches [AK13]. Instead of working directly with the raw CSI signal, they leveraged the CSI signals from two transmitters to cancel out the environmental noises and benefit from the phase information of the cleaned signal.

WifiU [WLS16] is a gait recognition system that uses an commercial off-the-shelf (COTS) WiFi devices to leverage the channel state information to capture fine-grained gait patterns for person identification. In contrast to expensive Doppler radars, the channel state information can provide similar information such as motion from echos caused by back-scattering from different body parts. *WifiU* consists of a router and a receiver to collect the modulated CSI due to human motions. A WiFi device sends continuously signals to its environment which are scattered by moving objects, such as a human within the transmission path. The scattered signals are then received by a laptop. A PCA based technique is used to reduce the environmental noise signals by extracting the principle components from the correlated CSI signals. The true movement results in dominant components within each sub-carriers and uncorrelated noise components can be suppressed by using the PCA method. After applying PCA, the time echo is still composed of reflections from various body parts. The decomposition of such a time signal can be performed by using a frequency-time spectrum (STFT) method. The reason of using Fourier transformation on the time signal is that different body part moves at different speed resulting in different Doppler shifts. The main goal is thus to transform the received CSI signals to the Doppler spectrum similar to other radar-based applications with high fidelity to extract Doppler motion information. Higher speed corresponds to higher Doppler shift and vice versa. Feature extraction is then performed on the cleaned Doppler shift profiles.

WiSee [PGGP13] is another application for sensing whole-body gesture recognition by leveraging wireless signals in an office environment or a two-bedroom apartment. Pu leveraged the frequency-time Doppler shift profile from various body parts while performing specific tasks, to achieve an recognition accuracy of 94 % on a set of nine gestures such as *push, pull, circle, dodge, drag, punch, strike, kick*, and *bowling*. Adib [AK13], developed by MIT researchers, showed various interesting use-cases by leveraging COTS WiFi devices. They were able to count persons, locate their relative positions, measure vital signs such as respiration rate and heart beat rate even from an adjacent room or behind closed doors. By treating a moving human as a moving antenna array, they were able to build an inverse synthetic aperture radar (ISAR) technique to enable radar-like vision. Thus they can scan the movement of the human in time by only using one single antenna.

WiFi-based activity recognition utilizes existing wireless transceiver infrastructure in the environment to measure activity induced WiFi signal variations. Compared to radar-based applications, WiFi application is more power efficient and preserves user's privacy, since no physical sensing module is required except the already existing WiFi communication route.

Discussion As reported by the cited works in this subsection, electromagnetic sensors are resistant to different weather conditions or other environmental noise at certain operating frequencies. In contrast to optical vision-based system, high frequency electromagnetic waves do not require a direct line of sight and can even penetrate through walls. In addition their robustness, safety, and reliability make them perfect to serve as an effective device for contact-free and ubiquitous motion monitoring of objects in the surrounding.

Due to its robustness against extreme weather conditions and large detection range, radar-based applications are already widespread in automotive sector for environment sensing and perception. Operating in the sector of HAR, the operating frequency and the transmit power should be reduced to adapt to indoor applications. Most use-cases work with radar sensors operated around 5.8 GHz, 7.25 GHz or 24 GHz. Human motions such as gait [SAZ19] or other whole-body interactions [KM15] can be leveraged to developed human-centered smart home appliances. Even sub-centimeter resolution of finger gestures can be observed with the specialized and miniaturized radar device *Soli* [LGK*16] integrated into a smartwatch or smartphone device.

For close range radar applications, UWB radar are often applied. Its advantages include low power consumption and more secure due to extreme short pulses, high transmission rate, and noise resistant due to ultra wide-band. Related to its superior physical properties, UWB can be used to perform exact indoor localization. The short duration of UWB pulses make them robust to multipath effects, since the identification of the main path from other multipath signals is more evident and thus allowing a more precise detection of the time of flight [ZGL]. Through the wall object imaging [LN05] is another useful ability of UWB imaging radar, especially in situations where a direct line-of-sight is not possible. For example, it can be applied in rescue operations or finding tracked person in a collapsed buildings.

WiFi application is more power efficient compared to general radar applications or UWB radars. Most WiFi-based applications work with modified WiFi access points. Compared to integrated hardware and software solutions of most radar applications, it is easier to modify the WiFi access points to adopt to specific tasks designed for HAR. Common applications build with modified WiFi devices are targeted at tracking and recognition of indoor activities. Close range applications include near device in-air hand gesture recognition [AYH15]. Room-scaled applications are commonly focusing on indoor localization [TKY*16] and tracking of human [AK13]. Based on Doppler profiles, whole-body gestures [PGGP13] can be targeted even when the sensor is placed behind the walls. Applied for localization tasks, the maximum detection range is up to 250 m outdoor and 35 m indoor [ZGL].

The usage of these sensor categories in the domain HAR is three-folds,

1. dynamic fine-grained whole-body activity recognition with radar-based sensors,
2. close-range fine grained activity recognition and imaging with UWB radar,
3. more power efficient whole-body activity recognition with WiFi signals.

Take-Home Message Radar applications are mostly used in outdoor environments with large operation frequencies, large detection range and high operating power such as environment sensing and perception of a vehicle on a motorway. Applications in indoor environments in case of human activity classification need to operate with lower frequencies and lower operating power. Most use-cases work with radar sensors operate around 5.8 GHz, 7.25 GHz, or 24 GHz. In case of CW radar or FMCW radar, a continuous signal is transmitted all the time, mak-

ing these applications less power efficient. For close range detection and sensing UWB radars are often applied due to its preferable physical properties.

However, most radar hardware are difficult to build. Commercial radar solutions have hardware and software packages strictly coupled such that an easy modification of radar software adapting to specific use-case is not accessible. One alternative is to use the channel state information of a commodity WiFi system. WiFi signals are easy to access and more power efficient compared to radar based applications, but operates at a much narrower operation frequency bandwidth of only 20 MHz compared to 1.79 GHz for a FMCW radar, resulting in lower time resolution than radar applications. An overview of the cited literature can be found in Table 2.6, where the previous works are introduced in terms of their application area, sensing device, processing algorithm, sensor behavioral, database and a concluding remark.

2.1.6. Other sensors and hybrid sensor-systems

Other physical quantities, such as temperature, chemical composition, and magnetic field modulation can be measured by dedicated sensors. However these sensors are not often used as a single sensing entity in the field of HAR [LL10]. Human activity is complex and it requires to capture information from multi-sensor networks to infer the correct actions [ZS09]. Variables such as temperature may add low level information to the process of activity reasoning, however, information fusion is needed to integrate the data in the high level decision making process. Magnet sensors can be placed on furniture, drawers, or doors to provide binary information when users directly interact with these objects [TBB09]. Temperature, light, pressure, humidity, or CO2 sensors are all components that can be used to build a wireless sensor network for smart home systems [RR15]. ZigBee [SM06], for example, is used as a low cost, low power, and less complex wireless communication standard to connect such sensor nodes with the main processing unit in a smart home system.

Applications integrating magnetic sensors into MEMS placed in initial measurement units (IMUs) are used for pose and acceleration measurement, mostly in wearable devices, such as smartphones, smartwatches, or other miniaturized on-body devices. Altun [AB10] used five body worn sensors placed on the chest, the arms, and the legs to classify daily and sports activities of eight subjects. Each sensor integrates a triaxial gyroscope, a triaxial accelerometer and a triaxial magnetometer. Combining feature dimension reduction techniques, such as PCA and sequential forward feature selection (SFFS) methods with Bayesian decision making classifier, they were able to balance between a high correct classification rate with relatively low computational cost with regard to real-time application.

Fusion multiple sensor categories to infer human actions is advantageous, because different sensor type provides different context (place, time, situation, etc). To ease the decision making process, a richer context is beneficial. Even combining multiple sensors from the same category, such as combining multiple acceleration-based sensors on the human body can increase the recognition accuracy of complex human activities. Maurer [MSSD06] investigated the classification accuracy of wearable sensor on different body position. Results demonstrated that the sensor placement strongly affects the recognition performance and could lead to misclassification if not properly placed.

Bao [BI04] revealed that two out of five bi-axial accelerometers were enough to recognize a set of 20 activities including ambulation and daily activities such as *scrubbing, vacuuming, watching TV,* and *working at the PC*. By only using the sensors on the hip and wrist as a sub set of all locations, the accuracy only decreased around 5 %. An increased accuracy of 25 % is achieved over the best performing single acceleration sensor. The fusion is performed on the feature-level by concatenating extracted raw features from the acceleration data time windows. However activities such as *stretching, scrubbing, riding escalator* and *riding elevator* were often confused. To overcome this issue, they required additional sensor modalities. Heart rate data can for example reveal the

Work	Device	Area of Use	Algorithm	Remarks
[LGK*16]	Millimeter wave radar	Fine-grained gestures	Bayesian inference	Motion resolution within sub-millimeter accuracy
[RLBL*18]	CW radar	Physiological signal	Majority voting, k-NN	Restricted study with 6 participants only
[LPRS12]	Dua Doppler radar	Fall detection	k-NN, SVM, Bayes	Aggregation by fuzzy integral, data acquisition from a trained actor only
[KM15]	Doppler radar with IQ-Demodulator	Military activities	DCNN	End-to-End training directly on the raw micro-Doppler spectrum
[SAZ19]	K-band radar	Identification based on Gait	Feature extraction with Prior PCA + Nearest Neighbour	Fuse features from both spectrogram and cadence velocity diagram
[WSL*16]	Millimeter wave radar	Fine-grained finger gestures	CNN + LSTM	End-to-end training
[TKY*16]	Wireless	Indoor Positioning	RSS, TOA, TDOA, FDOA	Survey
[AYH15]	WiFi RSSI	Hand gestures	Template matching, correlation	No additional modification to available wireless equipment or any extra sensors
[ZWX*19]	WiFi-based	Physiological signal	Maximize breathing-to -noise ratio, template matching	Signal processing makes it possbile to detect respiractory signal even through walls
[AK13]	WiFi MIMO, USRP software radios	Single, multiple person tracking through walls	MUSIC, to disentangle correlated, super-imposed signals	Moving target detection in the CSI channels
[WLS16]	Commodity WiFi device	Fine-grained gait patterns	Peak detection on the upper contour of torso movement from the spectrogram	Limitations on path and direction of the experimental setup, single person use-case
[PGGP13]	USRP-N210s	Whole-body gestures	Pattern matching with predefined templates	Preamble required, pre-defined template requires prior knowledge

Table 2.6.: Applications build on electromagnetic sensing.

intensity of physical activities and GPS location data can further provide the information whether the individual is at home or at work, and thus add a probability measure to certain set of activities.

Chernbumroong [CCY14] proposed a multisensor framework for activity recognition with genetic algorithm (GA) [GH88] to determine the fusion weights of the multisensor platform. The multisensor platform consists of accelerometer, temperature sensor, and an altimeter on a CC430F6137 Microcontroller with MSP430 CPU from Texas Instruments. Pressure sensor, gyroscope, barometer, and light sensor are integrated on Gadgeteer FEZ Cerberus board. In addition, a heart rate monitor is fixed on the chest with a chest strap. The sensor fusion is done both on the feature and decision-level (classification-level). To compensate for sensors that are less dependant in making decisions by themselves, such as altimeter and temperature due to their low-level context, these outputs are fused at feature-level to provide a richer context. The used feature selection was weighted by the feature importance. On the decision-level fusion, the outputs of multiple classifiers were fused using GA method to fine-tune the fusion weight parameters. The sum fusion on the decision-level improved the classification accuracy from 96.9662 % of the best single classifier to 97.3096 %. In 98 % of the experiment trials, the GA fusion method outperformed the one best single classifier.

Similar fusion methods were reported in the field of multi-biometric fusion [AHD*11], where other methods that take advantage of multi-decision coherence [DRBK17], variations in information source trust [DON14], or relative relation between confidence levels in multiple sources [DO14], can be mapped into the multi-sensor fusion in HAR applications.

Therefore, the context provided by one sensor category is limited. To infer complex human actions, richer context is required which can only be done by fusion of different sensor modalities. Integrating additional sensor or sensor categories can boost classification accuracy by achieving the following gains as reported in [Dam18] and initially defined by Bellot et al. [BBC02]:

1. Accuracy gain: accuracy of decisions and representations after the fusion process is improved. Noise and errors are reduced in comparison to single source information.
2. Completeness gain: the information after the fusion process is less redundant and more complete.
3. Representation gain: the information after fusion is more granular compared to each of the single fused sources.
4. Certainty gain: the belief in the fused information is increased.

2.2. Popular databases

In this section, we introduce several publicly available databases for the task of HAR, which are commonly used as baseline for researchers. They can be divided – based on our discussed sensor categories – into three groups: the single non-vision sensor category, the multiple sensor category, and the vision-based datasets. An overview of these databases can be found in Table 2.7.

2.2.1. Datasets using only one single sensor category

In the **Intel Research Lab** dataset [PFKP05], the authors used the RFID technology to recognize routine morning activities. They installed 60 RFID tags in the kitchen on objects touched by the user during a practice trial. The user wore two gloves built by Intel Research Seattle to detect that an object has be touched. However, unlike bar-codes, RFID tags cannot specify uniquely which instances of objects have been touched, rather that some objects have been touched.

Database	Sensor	Task	Remarks
OPPORTUNITY [RCR*10]	Hybrid sensor networks	ADL	large scale deployments of heterogeneous networked sensor systems
UCI DSADS [BY13]	multiple on-body accelerometers	ADL and Sport	dataset is restricted to wearable
PAMAP2 [RS12]	Accelerometer plus HR-Monitor	Physical Activity	enhanced basic context with additional sensor category
Amsterdam [vKNEK08]	Hybrid sensor network	ADL	upgrade-ability of sensor nodes, missing generalizability with dataset from one person
MIT PLIA [ILT*06]	Hybrid sensor networks	ADL	live-in laboratory
Intel Research Lab [PFKP05]	RFID tags	ADL	Single sensing category with binary information
Mocap [MRC*07]	optical marker-based	Motion clips	very clean and detailed motion capture data
Sports-1M [KTS*14]	vision-based	Sport clips	under unconstrained environment and contain complex activities
UCF101 [SZS12]	vision-basec	Sport clips	under unconstrained environment and contain complex activities
KTH [SLC04]	vision-based	sport clips	simple, isolated actions
MSRDailyActivity3D [WLWY12]	vision-based	ADL	action dataset of depth sequences noisy 3D joint positions if person stands close to a background object (sofa)
TUM Kitchen [TBB09]	Hybrid sensor networks	setting a table	tackles large pool of activites
URADL [MPK09]	vision-based	ADL	contains high resolution video sequences of complex actions, each containing only one specific task
CMU-MMAC [DlTHM*09]	Hybrid sensor networks	cooking, food preparing	limited variations due to small number of dishes
MPII Cooking Activities Dataset [RAAS12]	vision-based	ADL	fine-grained actions

Table 2.7.: Time Series Databases for Activity Recognition Tasks.

The UCI daily and sport dataset (**DSADS**) [BY13] is consisted of 8 subjects performing 19 different activities by wearing acceleration sensors on 5 body parts. Besides the more stationary classes such as *sitting, standing*, or *lying*, they also include dynamic exercises such as *ascending and descending stairs*, and *exercising on a stepper or a cross trainer*. However this dataset is only restricted to on-body wearable devices, even each wearable consisted of a gyroscope, an accelerometer and a magnetometer.

The **PAMAP2** dataset [RS12] targeted at physical activities such as *walking, cycling, playing soccer*, etc. It composed of 9 subjects performing 18 activities with 3 inertial measurement units and a heart rate monitor. Compared to **DSDAS**, this dataset fused another sensor category by integrating the heart rate monitor to provide additional information. As stated in [CCY14], fusion of several sensor modalities can provide richer context to improve the performance of recognition on more complex human actions.

2.2.2. Datasets using multiple sensor categories

Previous cited databases are either ubiquitous or wearable. However they only used one single sensing category and thus the provided context was limited. To overcome this limiting factor, other databases also use a composite of object sensors and ambient sensors to further incorporate more sensing modalities. The MIT **PLIA** dataset [ILT*06] were collected in a real experimental environment of 1000 sq.ft. apartment. *PlaceLab* is a new live-in laboratory for studying ubiquitous technologies in home settings. Approximately 214 sensors such as state sensors, accelerometer, camera, ambient sensors and object sensors were installed in the laboratory environment. During a 4-hour period, 89 activities are manually labeled from the collected sensor data.

The **CMU-MMAC** dataset [DlTHM*09] is another database leveraging multi-modal sensor data input for detecting tasks involving cooking and food preparing. Modalities collected were video, audio, motion capture, IMUs and two wearable devices. The dataset consisted of five subjects cooking five recipes, in average 15 minutes/recipe. In this database, people and objects were visually instrumented and thus making the videos less realistic. The limited number of only 5 dishes with very similar ingredients and tools lead to restricted data variances.

The **MPII Cooking Activities Dataset** [RAAS12] tried to close this gap of limited and constrained variations by providing a large database with more realistic, fine-grained activities. The database contained 65 different cooking activities performed by 12 participants. Instead of recording individual activity, the participants were asked to perform actions in sequence and recorded by video to reflect a more realistic behavior.

The **TUM Kitchen** dataset [TBB09] aimed to provide a comprehensive collection of sensory input data, to serve researchers in the field of marker-less human motion capture, segmentation and activity recognition. It collected of video data with four fixed overhead cameras, RFID tag readings and magnetic sensors detecting when a door or drawer is opened. All four subjects performed the same high level activity of setting a table. The dataset was constructed such, that it tackled challenges which is not covered in other available datasets. Those challenges are such as inter-class variability, change of human silhouette while interacting with objects, human performing several actions in parallel, occlusion by furniture, and subtle actions.

The **Amsterdam** dataset [vKNEK08] recorded the in house activity data of a 26 year old man, living alone in a three-room apartment monitored by 14 state change sensors placed different locations, such as on doors, cupboards, refrigerators, and a toilet flush sensor. Authors stated that the upgrade ability of their system was advantageous compared to other datasets [ILT*06] where sensors should be installed during the contraction time for intended locations especially build for research purposes. They claimed that if people are living in an unfamiliar environment, the action collected are not representative. Their solution was to leverage sensor network consisted of wireless network nodes to which simple off-the-shelf sensors can be integrated. In such way, they

can easily upgrade the user's living environment with wireless sensor networks. However, the dataset of only one person was limiting the results of its general validity.

The **Opportunity** database [RCR*10] was often used as a baseline dataset for HAR collected from wearable, object, and ambient sensors. It consisted of 4 users performing activities of daily living in an indoor environment. They deployed a wide range of 72 sensors of 10 different modalities in 15 wireless and wired networked sensor systems. The authors claimed that most existing datasets [ILT*06, vKNEK08] were not sufficient enough to investigate opportunistic activity recognition, where a large amount of sensors was required not only in the environment, but also on the body and in objects.

2.2.3. Vision-based dataset

Image based databases for HAR tasks are not rare. Datasets with constrained whole-body interactions, or datasets for outdoor sport activities are provided in [SLC04, NN08, WLWY12]. The **KTH** database [SLC04] currently contains 2391 sequences are collected under four different scenarios with 25 person performing six different activities, including *walking, jogging, running, boxing, hand waving*, and *hand clapping*. This dataset only included simple, isolated actions in staged data. No complex actions or multiple person case were targeted in this dataset. The data acquisition process was performed under constrained scenarios. This task of simple action recognition can be considered as "solved", since most techniques already report nearly perfect results [Pop10, TCSU08].

Compared to the **KTH** database, the **URADL** dataset [MPK09] contained high resolution video sequences of complex actions. It included 10 different activities such as *answer phone, chop banana, drink water, eat snack, look up in phone book, etc.* and were collected with high-resolution videos installed overhead. Even some classes were very similar, thus introducing more inter-class similarity, the scenes per video were constrained and each containing only one specific task. Fully unconstrained datasets in the wild were collected in [KTS*14, SZS12]. The **Sports-1M** is a database [KTS*14] collected from the web, containing 1,133,158 video URLs, which hast been automatically annotated with 487 labels. Also, the **UCF101** dataset [SZS12] consists of 101 action classes, over 13k clips and 27 hours of video data. This dataset contained user uploaded activities with unconstrained data collection process, containing camera motion and cluttered background. The unconstrained setting posed a challenging task for precise action recognition with computer vision methods.

Research in vision-based action recognition has made a lot of progress with the advances in deep learning and computer vision methods. Researchers moved on from recognizing simple, constrained actions to more complex actions or interactions with multiple person under unconstrained environments. Therefore, such databases containing unconstrained conditions and multiple complex scenarios, are considered to be more useful in this regard.

2.2.4. Discussion

Datasets with only single sensor category provided limited context and thus making it difficult to tackle more complex human actions. Therefore, databases composited of multiple sensor modalities or even the same sensing modality on multiple locations helped to solve more naturalistic and complex human actions. Common hybrid databases used composition of sensor modalities with low level information, such as state sensors, acceleration sensors, temperature sensors, and RFID tags. Image-based or video-based databases can provide rich context, however, often suffer from the problem of occlusion and privacy issues. If taken in private sectors, users may feel observed and thus do not act naturally or not representative of their usual behaviours.

Capacitive sensors or radar sensors can provide complex high-level information without violate the privacy. However, most of radar applications did not make their databases public. Ideally, a composition of these high-level information reasoned from capacitive, radar or WiFi sensors can be fused with low-level binary sensors instead of using vision-based systems, especially given the privacy concerns connected to vision-based sensors. The ability of these sensor to observe activities even through walls, makes them strong against occlusion and the line-of-sight problem. High frequency radar devices could resolve fine-grained action within sub-centimeter range and thus making the recognition of fine-grained and more complex actions possible.

2.3. Evaluation metrics

HAR can be treated as a pattern recognition problem, with the patterns related to specific actions. A list of the commonly used classifiers in the literature individually assigned to its categories can be found in Table 2.8. The most used classifiers and action detection methods in HAR can be divided in three large categories:

- Generative models: A generative model is a probability based method to learn the statistical distribution of the underlying data distribution. Generative model is able to create new samples according to the learnt statistics of the data distribution.
- Deterministic models: Deterministic models are static classifiers try to learn the hidden feature representations from the labeled training data. Discriminative model is intended to determine the membership of each sample to a certain class.
- Others: Other methods include non-parametric methods. Non parametric methods make no assumption of statistic distribution from the given data. They try to draw conclusions about the data from data with similar patterns.

Novel methods like the compressed sensing based HAR classification methods are currently drawing more and more attentions. These methods work with sparse representation and benefit from correlations in data to increase the processing speed and enable designers to place applications on devices with limited computing power. Examples of that are the works [CYL*17] and [CLG17] where the authors explored compressed sensing based HAR classification methods and achieved satisfactory results.

Evaluation metrics are necessary to compare different approaches and performances of action recognition systems. Though, the most metrics are defined for binary classification problem, they can be easily extended to fit multiclass classification problem. In this case, the multiclass problem can be divided into several binary classification problems. In Table 2.9, the most used evaluation metrics are given. As reported by Ward et al. [WLG11], a valid methodology for performance evaluation should fulfil two main criteria:

1. The metric should be objective and unambiguous. The outcome should not dependent on random assumption or parameters.
2. It should provide a quantitative measure to give a hint to the strengths and weakness of the system or method.

2.4. Discussion

Physical sensors are limited by its hardware and software characteristics. In the following, we discuss the hardware features related to the introduced sensor categories. We then identify some general challenges while performing software processing for these sensor categories.

Table 2.8.: Some of the most popular algorithms used for action recognition and classifiers in HAR along with examples of the works that utilized them.

Category	Abbreviation	Algorithm name	Source
Generative	HMM	Hidden Markov model	[AJLS97, WCS*16]
Generative	DBN	Deep Belief Network	[YNS*15, PHO11]
Generative	NB	Naive Bayesian	[MVPEL18, WCS*16]
Generative	GMM	Gaussian Mixture Model	[PK13, TN06]
Generative	denoise AE	denoise Autoencoder	[WCS*16]
Discriminative	SVM	Support Vector Machine	[GSC*17, GCC*19]
Discriminative	CRF	Conditional Random Field	[NDHC10]
Discriminative	LR	Logistic Regression	[ATSK12]
Discriminative	CNN	Convolutional Neural Network	[TBF*15, KM15, WSL*16]
Discriminative	LSTM	Long Short Term Memory	[WSL*16]
Discriminative	RNN	Recurrent Neural Network	[WSL*16]
Discriminative	ANN	Artifical Neural Network	[MPC16, SA19]
Others	k-NN	k-Nearest Neighbours	[GSC*17, GCC*19, OA00, WCS*16]
Others	DT	Decision Tree	[SVLS08]
Others	LDA	Linear Discriminant Analysis	[IMTP12]
Others	DTW	Dynamic Time Warping	[XHA*13, SCZ*14]
Others	FLD	Fisher Linear Discriminant	[QZK08]
Others	compressed sensing	Compressed Sensing	[CYL*17, CLG17]

Application	Metric	Definition	Source
Identification	top-1 accuracy	number of positives for all genuine match within the top 1 returned candidate list	[WLS16]
Identification	top-k accuracy	number of positives for all genuine match within the top k returned candidate list	[WLS16]
Classification	AUC	area under the receiver operating characteristic (ROC) curve the performance of a binary classifier system	[SVLS08]
Classification	Accuracy	ratio between number of correct predictions to total number of predictions	[WSL*16, XHA*13]
Classification	Recall	proportion of actual positives to correctly identified	[KSH07]
Classification	Precision	proportion of positive identifications to actually correct	[KSH07]
Classification	F1-score	a weighted measure between recall and precision	[SSSA12]
Classification	Confusion Matrix	indication of how well each class preforms and gets confused with	[AYH15, RLBL*18]

Table 2.9.: Some evaluation metrics commonly used in HAR along with examples of the works that utilized them.

feature	$--$	$-$	o	$+$	$++$
res	<8 m	<100 cm	<30 cm	<20 cm	<10 cm
upd	<1 Hz	<10 Hz	25 Hz	>50 Hz	>100 Hz
det	touch	<1 m	<5 m	<20 m	>20 m
unob	open large system	open small system	hidden, large exposure	hidden, noticeable exposure	invisible
proc	single sensor CPU	10+ sensors CPU	single sensor, embedded chip	10+ sensors by single chip	no further processing
calco	very hard	hard	normal	easy	very easy
sens	insensitive	less sensitive	normal	sensitive	highly sensitive
ls	<3 years	<5 years	5 years	>10 years	>15 years
wi	dependent	less robust	neutral	robust	invariant
fs	deformable	less stable	stable	robust	rigid
enc	highly sensitive	sensitive	normal	less sensitive	insensitive
occ	fatal	prone	neutral	stable	invariant
pe	>1000 mW	>750 mW	300 mW	<220 mW	<25 mW

Table 2.10.: Feature matrix denoting capabilities required for a certain rating. Features are graded in five levels, from ($--$, $-$, o, $+$, to $++$). List of Features are Resolution (res), Update Rate (upd), Detection Range(det), Unobtrusiveness (unob), Processing Complexity(proc), Calibration Complexity (calco), Sensitivity (sens), Life span(ls), Weather Dependency (wi), Form stability (fs), Electric noise coupling (enc), Occlusion (occ), Power Efficiency (pe).

2.4.1. Sensor hardware characteristics

Task-specific categorization of sensor selection is complicated. Tasks are diverse and are not limited to certain types of sensors. The appropriate sensor category to use is a design choice from the application designers related to user requirements. Sensor-driven categorization on the other hand is straightforward, as the sensing physical characteristics are fully describable and can be categorized. According to specific physical measures, the application designers are able to consider the appropriate sensor category.

Each sensor technology has its own advantages and disadvantages, limiting its use in various specific target applications. To better compare sensor categories to each other, standardized sensor specifications can be taken into considerations. In Table 2.10, we introduce some feature matrix denoting capabilities required for a certain rating. We grade the features into five categories, ranging from ($--$, $-$, o, $+$, to $++$). The scoring is derived from the research papers surveyed in this chapter and sensor specifications found from sensor data sheets. Some features depend on the use-cases and the form factor of sensor categories. Power efficiency for instance, is thus strongly dependent on the underlying system setup and not solely on the sensor technology. Similarly, the sensitivity is also a feature strongly related to how the sensor is applied in the specific system setup. Some of the discussed features are not quantitatively evaluated in previous works or are not measurable as a scalar. Therefore, we introduce our ranking for these features as a relative measure based on the description of the user experience. These features are, such as calibration complexity, weather dependency, form stability, electric noise coupling and occlusion. According to the assessment criteria presented in Table 2.10, the different sensor categories are graded in Table 2.11.

Acoustic sensors can work both contact-based or contact-free according to the specific task requirements. Contact-free sensors, such as microphones can classify human activities by leveraging acoustic events, but may

Sensor	res	upd	det	unob	proc	calco	sens	ls	wi	fs	enc	occ	pe
Microphone	+	++	++	-	-	−−	++	-	−−	++	++	++	+
Vibration Sensor	+	+	o	+	−−	-	+	+	++	o	−−	-	++
Ultrasonic Sensor	+	++	o	o	-	o	+	++	-	++	−−	-	o
Piezo-electric	++	-	−−	++	-	++	−−	−−	++	-	++	o	++
Fiber-optical	++	-	−−	++	-	++	−−	+	++	-	++	o	-
Capacitive	-	+	-	++	o	+	+	++	-	+	−−	+	+
Electrostatic	+	+	o	++	+	+	++	++	−−	++	−−	++	++
RGB cameras	++	o	+	-	o	-	o	−−	-	++	++	−−	–
Infrared imaging	-	+	o	-	+	o	-	+	+	++	++	−−	+
Radar	-	+	++	+	o	o	++	++	+	++	++	++	−−
WiFi	-	+	++	+	o	o	+	++	o	++	++	++	−−

Table 2.11.: Benchmark sensor system with respect to feature matrix given in Table 2.10.

raise privacy issues similar to a vision-based imaging system. Ultrasonic sensors on the other hand work in close range up to 5 m even in the darkness. Thus, it is invariant to illumination changes and weather resistant. However, since these systems are active, the power efficiency is worse compared to electric field measurement sensors, such as capacitance sensor or electric potential sensors.

Active capacitive sensing can work up to 15 cm in close range, but it is more noise prone, since noise detection in far range can not be resolved by the sensing system. Passive electric field measurement is purely passive and is sensitive up to 2 m in range. The passive measurement makes such system more power efficient. As the sensor is extremely sensitive to the ambient electric field, the system is prone to electric appliances or ambient powerlines. This requires additional hardware filters in the electronics design phase to reduce the powerlines coupling around 50 Hz.

Mechanical sensors respond to direct touch and are thus not susceptible towards powerlines and not susceptible towards ambient noise. Pressure signals are reproducible when the same force is applied, unlike electrostatic sensor which strongly depend on the varying ambient electric field. On the other hand, mechanical sensors are more susceptible to form stability. Especially, pressure sensors integrated into flexible textiles are prone to deformation. Deformation may easily break the pressure sensor or lead to performance degradation.

Vision-based systems are one of the most demanding research areas for HAR. With techniques using deep learning and large amount of online image resources, researchers are able to build robust segmentation and action detection algorithms. But the hardware limitations of the imaging system in visible spectrum, such as incapability of illumination resistance, occlusion, and change in object appearances over time, makes vision-based systems still a challenging topic.

Electromagnetic sensors are more resistant to environment coupling than any other previously mentioned sensor categories. They are robust against weather or climate changes operating at certain frequencies. They can cope with changing illumination or even occlusion cases, because at certain operating frequencies signals can even penetrate through walls. The hardware is designed such that the life span is long and form stability is high. To reduce the power consumption of radar-based devices, a modified WiFi access point can be leveraged to perform similar dynamic activity recognition tasks. Common commercial radar sensors with pre-designed hardware circuits are expensive and do not allow an easy modification of the signal processing in software with respect to a custom specification. WiFi devices, on the contrary, is already existent in the infrastructure and can be easily modified to gain access to the channel state information. The resolution accuracy of WiFi devices is indeed lower in comparison to high frequent radar applications, but with much reduced power consumption.

Therefore, how to choose the appropriate sensor category is strongly dependent on the design choice. According to range, obtrusiveness, robustness, and resolution, multiple sensor categories can be leveraged. Complementary sensor categories can be fused to provide richer context information to adapt to more complex human actions.

2.4.2. Sensor software characteristics

Regarding the software processing step, data-driven models extremely rely on the underlying data distribution. The performance is thus directly related to the data availability and data acquisition process. We identified some data-related challenges and software design issues encountered in the domain of HAR with sensor data. The following challenges are mainly divided into

- computation time,
- data acquisition process,
- database availability,
- data distribution,
- data augmentation ability,
- the intra-class and inter-class variability.

These aspects are considered to be important while designing a robust model to perform HAR with sensor data. In general, the process of data acquisition and the labeling task for HAR system are tedious and expensive. Extensive manual labelling and expert knowledge are often required. While image-based data are easy to acquire from the web or public databases, other non-visual data is less frequently available. There are several officially available databases with the focus on activity recognition for image or video data as introduced in Section 2.2. Images can be easily augmented using simple computer vision techniques, such as rotation, zooming, random cropping or applying noise filters to increase the amount of the training data. But it is not the case for time series. Time series are special, because the sequential information encoded in the time series can not be easily ignored. During the research phase, we identified that most of the applications with non-visual sensors collected their own database within a moderate test study and have not made it publicly available. Therefore, either unsupervised machine learning techniques should be applied to cope with the problem of missing labels, or shared database as benchmarks especially for time series data is desirable.

2.5. Summary

HAR is the key to enable human-centered application and enable natural interaction in a smart environment. To solve this challenge, the ability to learn the knowledge about human activity from raw sensor inputs is of critical importance. Therefore, in this chapter, we revised various research activities in this area and defined a number of sensor categories to perform this task. The overall choice of sensor category for specific application depends on various aspects. The practitioners should be able to make the correct decision conforming to the hardware characteristics of specific sensor types aiming to achieve the intended design goals. According to detection range, obtrusiveness, robustness, scale and resolution, multiple sensor categories can be leveraged. In this chapter I answered the research question 1 "*Which **sensor category** has to be applied under which conditions*" by providing a framework to make a comparison across the sensor categories possible with respect to certain HAR task.

Research field	Applications
Quantified-self	[GSC*17, SCZ*14, KAY*18]
Home behavior analysis (ADL)	[SWvHG11, TN06, ARK*06, CSZ*16, JHJP08, UNH08, SA19]
Gesture Recognition	[QHX*14, NITG16, PBRW*08, CMPT12]
	[SBQK05, LGK*16, WSL*16, AYH15]
Video Surveillance/ Analysis	[SBTM08, LL06, LWH03, LLO04]
Gait analysis	[PWQ*15, MPZN16, QZK08, GSD*13, LPRS12, WLS16]
Posture estimation	[BFMW15, LHL*13, RGPK14, XHA*13, RXS10, FMB*16, SFC*11]
Physiological signal sensing	[NGW15, PBRW*08, RLBL*18, ZWX*19, AK13]
Indoor Positioning	[STS*13, VMV09, FKvW*18, GPDH*16, BHH*13, TKY*16]

Table 2.12.: Research fields in human activity recognition with some common applications.

Application	Acoustic	Ultrasonic	Capacitive	Electric	Pressure	Camera	Radar	WiFi
Activity of daily life	x	x	x		x	x	x	x
Physiological signal		x	x	x			x	
Quantified-self					x	x	x	x
Postures Detection			x		x	x		
Gestures Detection		x	x			x	x	
Gait Analysis	x		x	x				x
Indoor Localization	x	x	x	x	x	x		x

Table 2.13.: Overview of the sensor categories used for each application in the domain of human activity recognition. We can identify missing application domains with certain types of sensor categories and future research directions.

Some surveyed applications with respect to the individual research field are grouped in Table 2.12. According to the surveyed most prominent research works in this chapter, I summarize in Table 2.13 the different sensor category used for certain applications in the domain of HAR. Given an illustration like this, it is possible to further identify missing application domains and provide some ideas for future research directions. I am able to identify existing research gaps to position my work and solutions developed in this thesis.

Finally, I identify several general challenges to be faced in this research field of action recognition with the previously introduced sensor categories. The main challenges can be categorized as follows:

1. Real-time detection, instead of offline processing: This requires smaller models, which can be applied on embedded devices with less computation powers. The capacity of the models should still be big enough to catch the underlying data representation.
2. Online-learning: Most of the machine learning models trained today are based on a fixed amount of training data and thus do not generalize well on new data. The ability to cope with new, unseen data, without the need to train the model again is thus a new requirement on the current model. The model should possess the ability of continuous learning.
3. Transfer learning and cross domain adaptation: The process of labeling HAR tasks is tedious and expensive. Therefore, if we can transfer knowledge from existing domain into a new domain with only less or mostly unlabeled data, it will save a lot of time and human resource of labeling.

4. Target the problem of inter-class and intra-class variability: Human motion is highly complex and possess a high degree of freedom. This can be expressed with the term user-diversity. Therefore, to design a robust model to cope with every possible situations, researchers should first target the problem of reducing the intra-class variability and increase the inter-class variability.

Again these challenges can be related to the research questions proposed in the introduction and are dealt with in this thesis.

- Challenge (1) is related to the research question 6. Possible solutions to this problem is introduced in Chapter 5 by manipulating the model capacity and minimizing the data processing efforts.
- Challenge (2) is related to the research question 5 and is partially answered in Chapter 4.2 by using methods to adapt the trained model on unseen data without retraining.
- Challenge (3) is related to the research question 4 and 5. The ability of transferring knowledge from relevant domains is especially explored in Chapter 3.1.7 and for the approach of domain adaption in Chapter 4.2.
- Challenge (4) is related to research question 5 and is targeted in Chapter 4.1. Diverse approaches are investigated aiming at covering the data diversity and improving the model performance towards a better generalization.

In the next chapter, I will introduce two mobile applications within the domain of Quantified-self using two novel sensor categories beyond the existing ones listed in Table 2.13. They are supposed to make contributions towards improved handling and performance for recognizing strength-based whole-body exercises.

3. Mobile applications

In the previous chapter we already discussed about sensor categorization and answered the research question 1 addressing at selecting the correct sensor category for certain HAR application domain. Connecting sensor categories to existent prominent research works in the sub-domains of HAR, I further identified research gaps to position my own work. This chapter revolves around the main issue of developing applications for Quantified-self with novel sensor categories beyond the existent common sensor technologies to further reduce the existing limitations on user handling and privacy. Through a detailed introduction of the individual design choices, system setups, processing and modeling, this chapter is intended to deal with the research questions 2 - 4 with respect to various issues encountered in the application design, such as data acquisition, data processing and data modelling for the classification task. I provide some common understandings and shared perception on common solutions to these research challenges in the summary section in this chapter.

Since the main focus is to reduce the requirement of wearable application commonly used in today's Quantified-self domain and permit the user to perform self-tracking anywhere at anytime, I emphasize on developing mobile applications with the ability of remote sensing. Mobile applications describe portable systems. Such systems encourage users to interact directly with the sensing interface. They are good for explicit interactions. Explicit interaction means, that users are intentionally interacting with the system. Mouses and keyboards are typical explicit interfaces, wherein direct intention of the user is transformed into instructions. The scope of my research here is mainly to develop novel and enhanced interfaces to better understand the user actions.

As the user is directly engaged in the interaction, there are certain aspects to be considered during the design phase. Instead of far-range detection, the emphasize is on the close-range detection and noise-free signals. The portability of such systems further requires them to be power efficient, form stable and resistant to different environmental changes.

Related to the design requirements, I identify two possible sensor categories to look further into. This chapter is grouped by the type of the physical measurands instead of task-oriented with the same justification as in Chapter 2, since the same task can be realized by different sensing technologies, each with its own advantages and disadvantages. This chapter is divided into two main sensing principles: active acoustic sensing with **ultrasonic sensing** and the active electric field sensing using **active capacitive proximity sensors**.

My contributions to these research fields are grouped as follows:

Mobile ultrasonic sensing

- Contribution 1: Exploratory study on the eligibility of using a commercial smartphone for remote activity sensing (Section 3.1.3)
- Contribution 2: Design mobile application for selected activity recognition with commercial smartphone using built-in hardware (Section 3.1.6)
- Contribution 3: Extended mobile and portable system for ubiquitous sensing at anytime on more realistic workout exercise set (Section 3.1.7)

Active capacitive sensing

- Contribution 1: Enable dynamic activity recognition with capacitive proximity sensors (Section 3.2.3)

In Section 3.1, I first explored possible use-cases by leveraging an unmodified commercial off-the-shelf (COTS) smartphone to sense its environment as an active sonar device. Subject to various explorations and observations, I identified several viable application scenarios of this sensing technique in smart environments, such as detecting basic activities of daily living like e.g. sleeping movement in bed, getting up and working activities on the desk, and mid-air gesture recognition. According to the experimental findings, I further identified the advantages of this sensing technique for recognizing whole-body interactions such as physical workout activities. In the preliminary user-study, I develop an application for detecting three distinctive physical exercises. This application is strongly limited to the sensor placement. The three activity classes are rather diverse and dissimilar. Therefore, the classifiers are able to find deterministic features. Finally, I further extend the application to loosen these constraints of sensor placement and increase the number of noticeable exercises to eight different classes, among which similar exercise classes coexist.

Based on various studies carried out on capacitive sensor applications, I identify that many applications using capacitive sensors are commonly applied on stationary tasks, such as posture recognition, physiological sensing or indoor localization. My second goal is thus to enable dynamic activity recognition with this rather stationary sensor information. In Section 3.2 of the active capacitive sensing, I extended a common yoga mat equipped with eight capacitive proximity sensors to enable tracking and counting of eight workout exercises for either person independent and person dependent use-cases. In order to compare across different sensing technologies developed in this chapter, the same set of exercise is selected.

3.1. Active acoustic sensing

Operating an ultrasonic sensor is by actively emitting high frequency sound waves around 20 kHz, which can be transmitted through air. This operation frequency is close to the upper audible limit of human being. According to the physical definition, ultrasonic range starts above the acoustic range and goes up to 200 MHz. It has the similar propagation property and wave characteristics as for electromagnetic signals such as radar or WiFi signals.

Raj [RKHD12] provided a survey on using Ultrasonic Doppler sensing technique in human computer interaction. Most previous works however focused on custom-built systems, instead of using consumer-grade devices directly without hardware modifications. A modern smartphone equipped with magnitude of sensors and processors, is more powerful than a working station in the 80s. Since a custom smartphone can sample audios with a sampling frequency of 44.1 kHz, we can use the in-build microphone of the smartphone to transmit a continuous signal with a carrier frequency of 20 kHz, thus turning the smartphone to an ultrasonic transducer. The following sections deals with the possible applications by using such a ultrasonic device without the need of integrating extra hardware.

The use of ultrasound as a sensing modality has been widely investigated in the research literature. Filonenko [FCC10] explored the feasibility and limitations of ultrasound sensing using mobile phones. They evaluated the performance of several mobile devices on generating signals with a frequency range of 17-22 kHz. A basic finding is that none of the test devices met any difficulties of generating inaudible sound frequencies. This makes mobile devices for indoor positioning system using ultrasound trilateration a viable option. Recent advances in using a commercial off-the-shelf device to perform activity recognition has become more popular. Following application areas are identified through a literature review on this topic.

Gesture Recognition Detecting finger and hand movements in free air is often achieved by analyzing a backscattered ultrasound signal. Kalgaonkar [KR09] applied ultrasonic waves to unobtrusively recognize one-handed gestures on custom hardware. The authors employed the Doppler effect caused by a moving finger in the vicinity of the sensing device. Here, a single transmitter and three receiver microphones are sufficient to

determine a 3D movement. Due to the availability of ultrasound capabilities in consumer hardware, ultrasound approaches have also been ported to consumer smartphones and laptops. SoundWave [GMPT12] is a hand gestures recognition system by using an unmodified consumer-grade laptop. Another similar application is Dolphin which detects various hand gestures performed above a consumer smartphone using the system's internal speaker and microphone [QHX*14]. Although the Doppler measurements are not discriminant in theory, the authors argued that left- and right-swipe gestures can be classified on a per-user basis.

Activity Recognition, Location and Context Awareness In order to detect whole-body movements, worn nodes as well as passively backscattered signals can be used. Multiple microphones enable ranging and localization using the angle or time-difference of arrival. Tarzia [TDDM09] used a similar technique to determine user presence near an unmodified consumer laptop. By using a measurement window of only 10 s they achieved an accuracy of approximately 96 % to discriminate two classes: absence and presence. By further extending the windows to 25 s, they achieved almost perfect accuracy.

Using custom hardware, Kalgaonkar et al. [KR08] were able to detect whether a person talks in front of an ultrasound Doppler sensor. Sound propagation on human skin can be applied to measure touches and gestures with combined hardware worn on the finger and arm [MCT*13]. By attaching ultrasound emitters to various parts of the body and a transceiver worn around the neck, Watanabe, Hiroki and Terada et al. [WTT13] are able to derive various activities such as *sitting* and *cleaning*. Rossi [RSA*13] measured the impulse response of different environments with a mobile phone to infer an indoor location. The authors were able to distinguish between more than 20 rooms. Similarly on a smaller scale, emitting sound and vibrations can be used to excite the underlying surface of a device and thus enable localization [KL07]. Very recently, the authors of [NGW15] presented an approach to recognize sleeping apnea using unmodified mobile phones to measure the user's chest movement in sleep.

Multi-Device Interaction Ultrasound messaging and ranging is a very convenient way to identify and track multiple devices. Techniques which use simple backscattering are not able to identify objects and thus leave a certain amount of ambiguity in their results. By embedding information in emitted ultrasound signals can thus making object and person localization and identification [AGG*13] possible. By combining wireless network and microphone interfaces of mobile devices, a more reliable calculation of the absolute time-of-flight of the ultrasound signal between two devices were possible, as proposed in related works [BLO*05, SHS01, HWL*03].

Approaches for device selection on mobile phones, e.g. by exploiting the Doppler effect when pointing at a device, has been investigated in [RM01, SPBZ13]. BeepBeep [PSZ12] solely relies on ultrasound generated by smartphones, which supports ranging using the time difference of arrival and localization. In case more than three objects are involved, it is possible to determine a relative position leveraging the information of multiple devices. Sensing device location in a car was investigated in [YSC*11]. By classifying the mobile phone's position it is possible to differentiate between the driver using the phone or a passenger. Reynolds [RMD07] introduced an ultrasound position sensing system for tangible objects above LCD-screens. The authors presented interactive 'pucks' that communicate by emitting and receiving ultrasound to reconstruct their position.

In contrast to the typical use-cases of ultrasonic sensing in the industry, such as level or leakage measurement or as an imaging system in the medical domain, I am more interested in using this technology for activity recognition tasks without the need to design any additional hardware circuits. Leveraging multiple device for recognition and tracking is impracticable beyond the laboratory setup, therefore in the preliminary study I aim at using only one single device to construct the desired application. This subsection is organized as follows: I first introduce an exploratory study to showcase the possibility of modifying the smartphone to continuously emit and receive ultrasonic signals of 20 kHz. I then investigate several possible use-cases which can be developed using this technology. I further identified possible missing use-cases and build real applications in the later researches, such as recognizing from simple sport exercises to more complex and more realistic workout exercises.

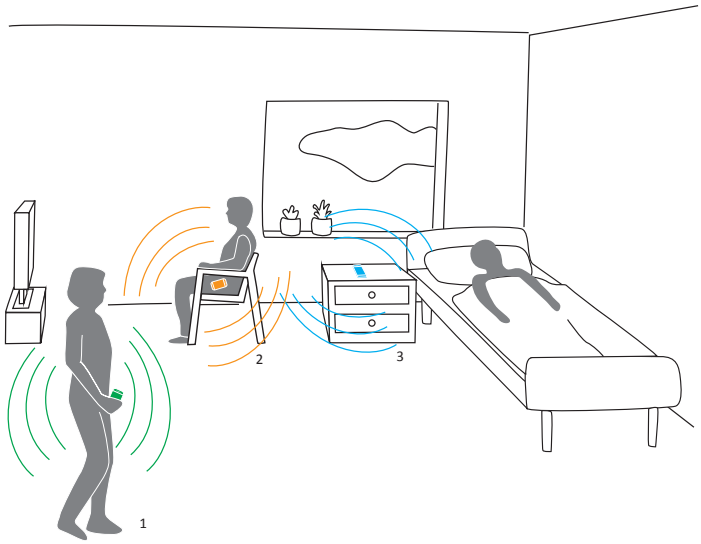


Figure 3.1.: Ultrasound sensing using the smartphone's native speaker and microphone can be used to capture information in various scenarios: (1) holding the phone, (2) carrying the phone on the body, and (3) stationary deployments.

3.1.1. Introduction

Streaming classification techniques are widely used in mobile devices to recognize human behaviours and contexts. This is extremely valuable to realize implicit interaction systems, for example to support healthy and independent living. The most important parameters to sense include indoor location, gestures, heart rate, or emergencies like falls.

Mobile devices bring along many sensors that can be used for this purpose. However, interaction is mostly centered around the device and information from the distant environment is harder to capture with integrated sensors. Besides that, the applicability of certain sensors can be limited by the use-case. For example, camera-based systems induce privacy issues and body-worn systems are sometimes inconvenient to wear over long periods.

Here, we investigate the use of ultrasound to support new, unobtrusive, sensing possibilities on the mobile phone to overcome these issues. We present initial experiments that outline opportunities for activity recognition in wearable and stationary settings with our sensing technique. In our exemplary implementation we emit an ultrasound wave with the mobile phone's speaker. The phone's microphone picks up the reflected signals and enables to derive information on objects at distances up to 2 m. In order to demonstrate the capabilities of ultrasound sensing, we use continuous-wave with a center frequency at 20 kHz.

In the targeted scenario, a consumer-grade smartphone could potentially avoid the need for additional hardware equipment. It could realize new use-cases, such as detecting falls while placing the smartphone on the night desk. This provides the consumer with a simple-to-execute solution that can be installed conveniently as a single mobile application. However, there are numerous challenges which are mainly induced by the heterogeneity of target systems. Firstly, this comprises the use of low-cost to high quality microphones and speakers. Secondly, the target system's software and hardware capabilities are very different in terms of processing power and battery

run-time. These challenges are considered in our exploratory study to increase the awareness of the application designers.

3.1.2. Physical principles of Doppler sensing

Currently established range and movement measurements often rely on pulsed radar. Here, a short pulse or a burst of short pulses is generated and the reflections are measured. This technique grants inferring distance measurements to nearby objects. Other methods emit a continuous wave (CW) with a fixed frequency or a frequency modulated continuous wave (FMCW). The first technique permits for recognizing Doppler effects caused by movements, while FMCW also provides static distance measurements. In contrast to FMCW and pulsed methods, simple CW does not induce demanding timing constraints and is easily realizable on a wide range of smartphones. For this reason, we use continuous wave modulation in our experiments.

A continuous wave, which can be a pure sine-wave with a frequency f_0 is sent out from the system's speaker. We use the low ultrasound frequency of 20 kHz, which is just above the human audible limit that can still be achieved with the phone's hardware components. This operation frequency is selected bound to the maximum audio sampling frequency of 44.1 kHz by the smartphone. The maximum audio frequency is considered sufficient to sample Doppler shift around this central carrier frequency.

A reflection from a moving target broadens the frequency spectrum around the central carrier frequency, which is called the Doppler frequency shift. An approaching target induces a shortening of the received wavefront, which means the frequency increases and a positive Doppler shift is observed. Respectively, a departing target leads to an expanded wavefront with a negative frequency shift. In order to measure the Doppler frequency, Fast Fourier Transform (FFT) is used. The number of samples N_{FFT} used by the Fast Fourier Transform (FFT) determines the individual bin width of Δf. Therefore, it also determines the resolution of the measurable Doppler frequency.

Sampling an audio signal using the in-built microphone of a smartphone can usually be conducted at a maximum frequency of $f_s = 44.1$ kHz. Regarding the Nyquist theorem, the highest retrievable frequency for an ultrasound measurement is thus $\frac{f_s}{2} = 22.05$ kHz. Since the human ear is typically not able to perceive sounds at frequencies up to 18 KHz [FZ07], this leaves us with an effective frequency range of 4.05 kHz (22.05 kHz - 18 kHz), corresponding to a theoretical maximum relative speed of $17\frac{m}{s}$. In order to obtain a reasonable temporal resolution for gestures and movements, we divide the signal into overlapping windows (50 %) with a length of 93 ms containing 4096 time samples, resulting in a time resolution of 46.5 ms due to the 50 % overlap.

$$t_{win} = N_{FFT} \cdot (\frac{1}{f_s}) = 4096 \cdot (\frac{1}{44100}) = 93ms \tag{3.1}$$

This corresponds to a $N_{FFT} = 4096$-point FFT with a bin resolution of 10.75 Hz.

$$\Delta_f = \frac{\frac{f_s}{2}}{\frac{N_{FFT}}{2}+1} = 10.75Hz \tag{3.2}$$

With a carrier frequency of 20 kHz, Doppler-shifts are observable with a resolution of 0.09 $\frac{m}{s}$. In comparison, gestures in front of laptops were observed with a maximum speed up to 3.9 $\frac{m}{s}$ [GMPT12].

$$\begin{aligned} f_d &= \frac{2 \cdot v \cdot f_0}{v_0} \\ v &= \frac{f_d \cdot v_0}{2 \cdot f_0} \\ &= \frac{10.75Hz \cdot 340\frac{m}{s}}{2 \cdot 20kHz} = 9\frac{cm}{s} \end{aligned} \tag{3.3}$$

The Doppler equation is given in Equation (3.3). By using the minimum frequency resolution $f_d = 10.75Hz$, we can easily calculate the observable speed resolution of 9 $\frac{cm}{s}$, by rearranging the Doppler equation. The f_0 term in the Equation 3.3 represents the central carrier frequency of the emitted signal and the speed v_0 indicates the approximated speed of the ultrasonic wave propagation through air.

A spectrogram is depicted in Figure 3.2. This is a 2D image of the spectral amplitude of frequency over time. For each time window of 93 ms, a hamming window is applied to the time window to reduce the leakage on the boundary of the time segment. Then a $N_{FFT} = 4096$ point FFT is calculated. The one-side of the two sided amplitude distribution over the FFT bins is collected, since the FFT is calculated over a real-valued time series. Then the time window moves to the next 93 ms with and overlap of 50 % to build the next FFT and so forth. The result is the 2D spectrogram with the signal power given in the unit of dB. This method is called the short time Fourier transformation (STFT).

The STFT of the discrete input time signal is calculated by Equation 3.4. It depicts the spectral distribution of a time series in the frequency-time domain.

$$X_m(\omega) = \sum_{n=-\infty}^{\infty} x(n)w(n - m \cdot R)e^{-j\omega n} \tag{3.4}$$

where

- $x(n)$ = discrete input time signal at discrete time instance n,
- $w(n)$ = length M window function (e.g. hann or hamming) to reduce the leakage on both window boundaries,
- X_m = discrete time fourier transformation (DTFT) of windowed data centered about time $m \cdot R$,
- R = hop size, also means overlap, between successive DTFT windows.

The power spectrum is then given by the Equation 3.5.

$$X_{STFT}(\omega) = 20 \cdot log(|X_m(\omega)|) \tag{3.5}$$

In our later work, we will see the influence of the number of FFT points in frequency domain on the resolution in time and vice versa. This is called the time frequency uncertainty principle [Coh95]. Higher resolution in frequency domain means more sample points in time are needed to calculate the Fourier coefficients. A larger observation window in time leads to worse resolution however. Vice versa, by applying shorter time windows, we will lose more fine-grained information in the frequency domain, thus introducing more quantization error in the speed range. Therefore, in case of more complex and slower whole-body exercises, a coarser resolution

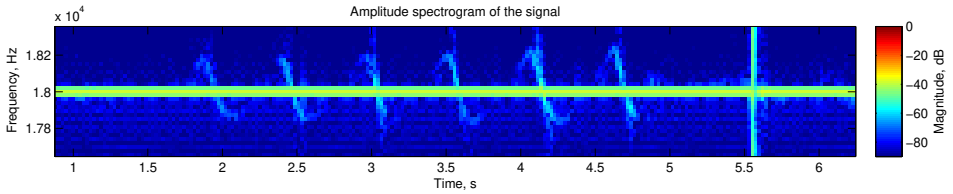
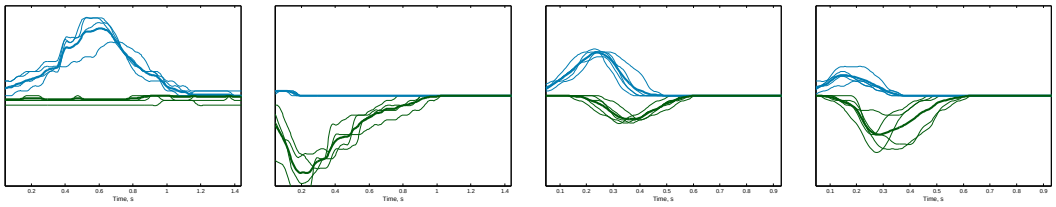


Figure 3.2.: An exemplary spectrogram: The Doppler-shift is caused by a waving hand in distances between 0.1 and 0.3 m atop the smartphone. At second 5.5, the user claps both hands, resulting in a large noise overlay. Here we used a signal with a center frequency of 18 kHz.



(a) (b) (c) (d)

Figure 3.3.: Broadening of the spectrum around the central carrier frequency. Thin lines indicate individual experiment runs, while the thick line represents the mean of all runs. In addition, blue colored curves depict the broadening due to a positive Doppler shift while the green curves represent the negative shifts. (a) Downward motion (b) Upward motion (c) Strong swipe motion: Right-to-left for right hand or Left-to-right for left hand (d) Weak swipe motion: Right-to-left for left hand or Left-to-right for right hand

in time should be preferred, in order to make the resolution in frequency fine enough to recognize slower body movements. This uncertainty issue thus makes this time-frequency relation yet another very important parameter to tune.

In the following experiment, we are only interested at the silhouette of the Doppler broadening extracted from the spectrogram around the carrier frequency of 20 kHz and thus reducing the storage demand on the sensing device. This Doppler profile already enables us to extract sensing characteristics and physical properties. As depicted in Figure 3.3, the upper curve presents the positive Doppler broadening indicating movement towards the sensing device, while the lower curve presents the negative Doppler broadening, indicating movement away from the sensing device. The higher or lower the Doppler curves are, the faster is the relative speed towards the sensing device. The absolute energy in the spectrogram is not considered in the exploratory study.

3.1.3. Experiments

This experimental section is based on our work in [FKGP*15], where we thoroughly examined the feasibility of the hardware and system performance. This is our first exploratory study intend to show the feasibility of turning a smartphone to an ultrasonic device to detect human activities in general without additional hardware. Similar applications by leveraging the personal computer to emit and receive ultrasonic waves of 20 kHz to track mid-air hand gestures are already proposed by several other researches. However, by the time of our research work, the

use-case of turning a commercial smartphone into an active sonar system without hardware modification is still relatively new. We structure our first research work in this direction as follows:

1. Perform various experiments verifying the hardware limitation and the sensing characteristics of this sensing technology
2. Conclude possible use-cases and provides future research directions
3. Perform mid-air gesture recognition with commercial smartphone

In this subsection, we present various experiments leveraging the above mentioned technique and discuss their benefits and limitations. We analyze what kind of experiments can be conducted using a stationary and non-stationary deployment of a mobile phone. These include the recognition of hand gestures during which the user actively interacts with the phone, as well as passive interaction when determining activities in the vicinity of the device. Additionally, we investigate the performance of our approach when executing gestures while holding the phone as well as while wearing it on the body, such as in the trouser pocket.

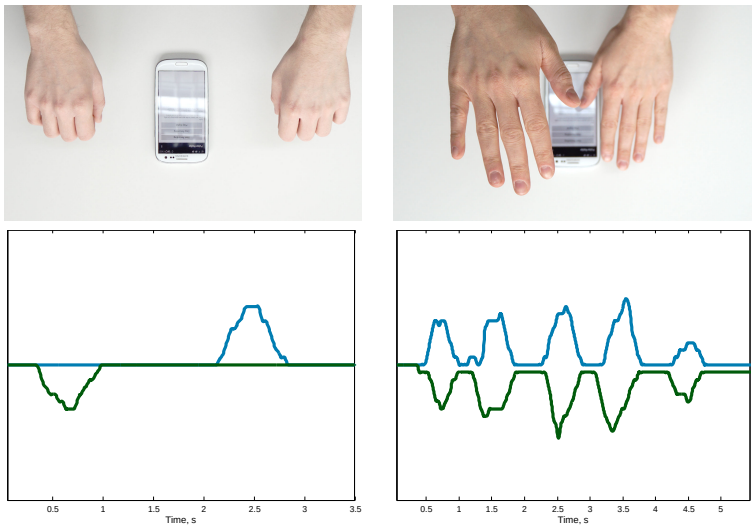
The first part of experiments shows how the Doppler shift can be used in a controlled setup and whether it is feasible for gesture and activity recognition. The second part aims to test real-life scenarios where we expect a noisy signal. These tests will show if our approach can overcome those obstacles. We conducted the experiments on a ZTE Blade, an Asus Nexus 7 and a Samsung Galaxy S3 running Android 4/5.

3.1.3.1. Stationary Deployments - Gesture Recognition

For the first set of experiments, the phone is positioned on a table, while the user performs gestures above the phone. Due to the Doppler frequency shift caused by the user's hand movement, we can determine whether the hand approaches or withdraws from the device. For the downward motion, we expect a positive shift in frequency, while an upward motion will cause a negative shift. This effect can be qualitatively measured by thresholding the frequency amplitude and calculating the signal envelop in both directions around the center frequency in each time step. An example for this effect is illustrated in Figure 3.3 displaying the broadening of the spectrum around the central carrier frequency for various gestures. Note that for these kind of diagrams, we deliberately abstained to quantize the vertical axis to direct the focus on a qualitative interpretation of the results.

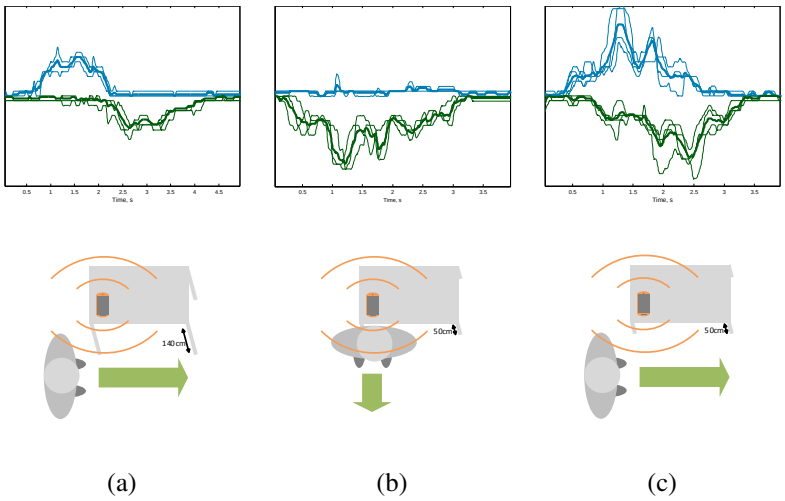
A central limitation of recognizing gesture via the Doppler Frequency shift in combination with consumer mobile phone is the fact that we can only determine the relative change in distance of an object to the device. For example approaching the phone with the user's hand can be done in any arbitrary angle between the table and the velocity vector of the hand. As long as the euclidean distance between the hand and the phone changes at the same rate, one cannot distinguish these motions by means of the Doppler Frequency shift. Figure 3.4(a) illustrates this challenge. At the start of the experiment the user places both hands slightly above the phone and then does a zoom out and zoom in movement by moving the hands into opposite directions and back together again. The resulting shape is similar to a combination of up- and downward motions as depicted in Figure 3.3(a) and 3.3(b).

However, slight variations in auxiliary movements can be picked up by the system. For example, the movements of the arm when doing a swipe gesture above the phone. Due to the nature of the Doppler effect swiping from left-to-right should be equal to swiping from right-to-left, hence these motions should not be distinguishable. This holds true for the hand movement, but not for the motion of the arm. Doing a right-to-left swipe with the right arm is more natural than doing the opposite swipe direction with the same arm. We performed several experiments using swipe motion with both arms and concluded that it is possible to distinguish the strong swiping motion from the weak one. In this scenario, a strong swipe motion would be the natural movements for the respective arm with the favorable direction, right-to-left for the right arm and left-to-right for the left arm.



(a) (b)

Figure 3.4.: Broadening of the spectrum for two-handed gestures. The blue curve indicates the positive Doppler broadening representing movement towards the sensing device, while the green curve indicates the negative Doppler broadening, representing movement away from the sensing device. The higher or lower the Doppler curves are, the faster is the relative speed to the sensing device. (a) Zoom out and zoom in gesture. (b) Seesaw motion, including up- and down motions at the same time.



(a) (b) (c)

Figure 3.5.: Broadening of the spectrum for different motions. (a) Walking by a phone at chest-height (b) Walking away from a phone at knee-height (c) Walking by a phone at knee-height

The weak swipe motion follows analog: The results are presented in Figure 3.3(c), 3.3(d) and depict the motion of the hand coming closer to the phone and going away again. A swipe motion from right-to-left with the right hand can be split into a fast (larger positive frequency shift) approaching motion and a slower (smaller negative frequency shift) withdraw motion. Generally speaking, the motion part that happens near the respective arm is executed faster and makes it easier for the model to distinguish strong and weak motion. In an application scenario where swipes are executed using a predetermined hand, one can in fact discern between different swiping directions.

An important advantage of our approach is the possibility to recognize multiple different motions within the same time frame. Executing a motion towards and away from the phone at the same time results in two separate frequency shifts in opposite directions. In this experiment two hands are used to execute a downward and an upward motion at the same time. Upon reaching the phone with one hand the respective motion are reversed. We call this the seesaw motion. Figure 3.4 (b) illustrates this scenario and shows the respective positive frequency shift for the hand going downwards as well as the negative frequency shift for the hand doing the upward motion. In the end both hands come to rest in a middle position at the same height resulting in a lower frequency shift due to the slower movement.

3.1.3.2. Stationary Deployments - Motion Recognition

Apart from recognizing hand gestures, we believe that the system is able to perceive general motion or activities in the vicinity of the mobile phone. To test this hypothesis we mimicked our swipe motion during gesture recognition. In this experiment, the phone is deployed on a table at chest-height and the subject simply walks by the table. The resulting graph, shown in Figure 3.5(a), clearly depicts a motion towards and away from the phone. This experiment was designed to mostly pick up the motion of the upper body of a person. Although having a similar appearance as Figure 3.3(c) (strong swipe gesture), both actions can clearly be distinguished by including the duration of the movement. In particular, walking by the phone takes almost five seconds, while a swipe is generally executed way faster at around one second.

To investigate whether the motion of individual body parts can be detected as well we conducted another test run where the phone is placed on a lower table, approximately at knee-height, while the subject walks away from the phone. While the person is increasing the distance to the phone, the broadening of the spectrum should show an overall negative Doppler shift, which can be seen in Figure 3.5(b). It also shows multiple peak shifts overlaid to the overall negative Doppler shift corresponding to the movement of each leg. These are stronger the closer a subject is to the device, as the reflections are easier to be picked up in close proximity.

Mimicking our first test in this series, we redid the walk-by experiment with the phone placed on the lower table. We expected to pick up a stronger factor for the movement of individual body parts, like the legs, contrary to our first test, where the phone was deployed on a table at chest-height. The graph is shown in Figure 3.5(c) and confirms our presumption. Clearly visible are spikes of Doppler shifts caused by the fast movement of the legs in comparison to the overall speed of walking direction. Surprisingly, it seems that both positive and negative part of the Doppler motion (approaching and withdrawing) has been shifted closer together. No apparent transition is visible as in Figure 3.5(a). We conclude that the device registers the motion of the upper body just as in the previous experiment. However, this motion may get smeared by individual leg motion, e.g. when the body still approaches the phone, but one leg already past it.

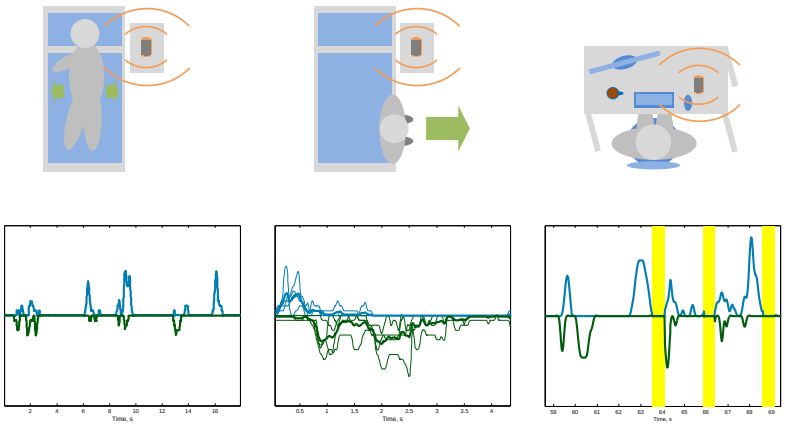
3.1.3.3. Stationary Deployments - Activity Recognition

In this series of experiments, we aim to analyze more complex activities with the gained knowledge. For the first test, the user is laying on a bed and the mobile phone is deployed on the nearby bedside table (at knee-height). We want to know whether our approach can perceive subtle movements of a person while asleep. By doing so, one could analyze the sleeping rhythm of the user which could be beneficial for estimating the best time to sound an alarm clock. Figure 3.6(a) shows the broadening of the spectrum for a person moving around on the bed, e.g. the subject changes from a face-down lying position into a dorsal position. These motions are being picked up by the device as Doppler shifts.

We want to combine this experiment with our motion experiment by letting the subject get up from the bed and walk away from it. The broadening of the spectrum for this experiment is shown in Figure 3.6(b) and is consistent with the executed motion. At first, we observe a positive shift due to the subject standing up, hence decreasing distance to the phone. When walking away, we register negative frequency shifts and several peaks when the subjects moves each leg. Additionally exaggerate arm movements may also be picked up.

In a last test run, we investigate our approach in the context of everyday desk work. The user is instructed to work with a computer at the desk as usual, while we observe the signal received by the mobile phone, which is also placed on the table. For simplicity, we only show a part of the whole recording in Figure 3.6(c). The graph depicts various positive and negative Doppler shifts corresponding to hand and/or mouse movements. In the spectrum itself, we observed large peaks over the whole frequency spectrum indicating noise produced by the user. In particular, pressing keys on the keyboard may result in such sounds. To better differentiate these from the measured Doppler shifts, we highlighted these peaks with marked labels in the figure. In certain cases the peaks come paired with positive and negative Doppler shifts. In our scenario, the subject was drinking from a cup of coffee. The movement of picking up the cup and placing it on the table can be related to such a signal response.

This series of tests indicate that the system is able to perceive general motions in the proximity as well as the movement of individual body parts. Given an appropriate data set and calibration, previously trained activities can be recognized.



(a) (b) (c)

Figure 3.6.: Broadening of the spectrum for different user activities. (a) Sleeping Movement in a bed (b) Getting up after sleeping (c) Excerpt of desk work. The color yellow marked labels represent broad frequency noises caused by short event pulses in time, such as placing a cup on the table.

3.1.3.4. Stationary Deployments - Range Limitation

This leads us to the question: What is the maximum distance to the smartphone at which we can reliably detect a Doppler shift, considering the noise from the received echo signal? The setup uses a ASUS Nexus 7 placed on top of a shelf with a height of 140 cm. This corresponds to the chest height of an average man. In the experiment, test subjects were asked to approach the device from different distances. We started at 250 cm away from the device and instructed the subject to get closer in each following step. The step width is chosen to be 50 cm. We stopped the recording as soon as we reached the device. The result is depicted in Figure 3.7 and shows the spectrum of the received signal over time. However, it should be noted that the signal is filtered with a digital notch filter at the center frequency of 20 kHz, so that the main component at this center frequency coupled back from the device is reduced. The first peak appears at a distance of 200 cm away from the device and corresponds to the first step taken from 250 cm. The following peaks are separated 50 cm from each other. Therefore, the last peak is taken directly in front of the device. It should also be mentioned that in order to increase the clarity of the plot, only frequencies above 20 kHz are shown. Since we are approaching the device, only positive Doppler shift have to be evaluated.

3.1.3.5. Non-stationary Deployment - Holding the Phone

Contrary to stationary deployment of the device, holding it can provide an auxiliary input method via simple gestures. For example up- and downward motions of the other hand in front of the phone can be used to control the volume of a music track or zooming into a map view. Swipe gestures prove useful when sorting through catalog, like a photo gallery or the mail folder. However, movements are limited to one-handed gestures, while stationary deployment can make use of two-handed gestures or whole body movement. We executed test runs using a wave-like (up- and downward motion) and a swiping gesture and successfully extracted the spectrum broadening as shown in stationary experiment. Nevertheless, while holding the mobile phone in one hand,

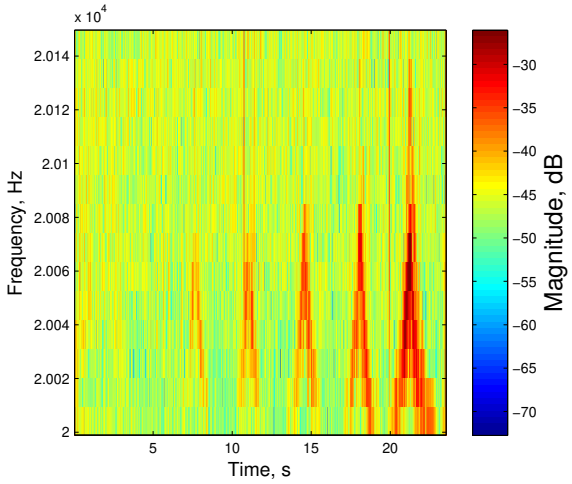


Figure 3.7.: Spectrum of a user approaching the ASUS Nexus 7 from certain distance is depicted here. The first peak at time instance of 7 s corresponds to a distance of 200 cm away from the mobile device. The following peaks are separated 50 cm from each other till the person is standing right in front of the device.

we detected an increase in noise around the central carrier frequency and the shifted signal. This is due to the unavoidable movement of the hand carrying the phone, causing very small, irregular frequency shifts and thereby deteriorating the received signal.

We want to find out to what extend the movement of the phone itself influences the received signal and prevents a reliable detection of gestures. In this series of experiments we vary the amount of 'noise movement' while performing a wave-like gesture (up- and downward motion). As a baseline we perform the gesture while the phone rests on top of a table (Figure 3.8(a)). Additionally we execute the same gesture while holding the phone in one hand (Figure 3.8(b)) and while walking around (Figure 3.8(c)). These experiments show that moving the phone itself while recording gestures induces a lot of frequency shift noise and hinders gesture recognition quite a lot. Possible solutions would include the estimation of 'walking noise' and subtract it later on to clean the signal. However, this is only suitable while walking in wide areas. Narrow corridors, for example, may cause unintentional frequency shift when walking by an open door.

3.1.3.6. Non-stationary Deployment - Carrying the Phone on the Body

In this section we conduct experiments with mobile devices carried on the body. In the first stage, we placed the mobile phone inside the clothing. It was carried in the pocket of a thin trouser. The device was put into the pocket after the recording was started. Figure 3.9 shows an extraction of the recording, where a hand is approaching the mobile device and moving away again. The same action was repeated for three times. Afterwards the recording was stopped. However, it can be seen in Figure 3.9, that through the thin material, noise covers most of the echo signal. This effect hinders the reliable recognition of gestures. For thicker trousers, like jeans, no frequency shifts can be registered at all.

In the second stage of the experiment, we strap the mobile device to a runner's arm to measure his surroundings while jogging. We hope to explore possibilities of distinguishing the users' environment, like e.g. urban area,

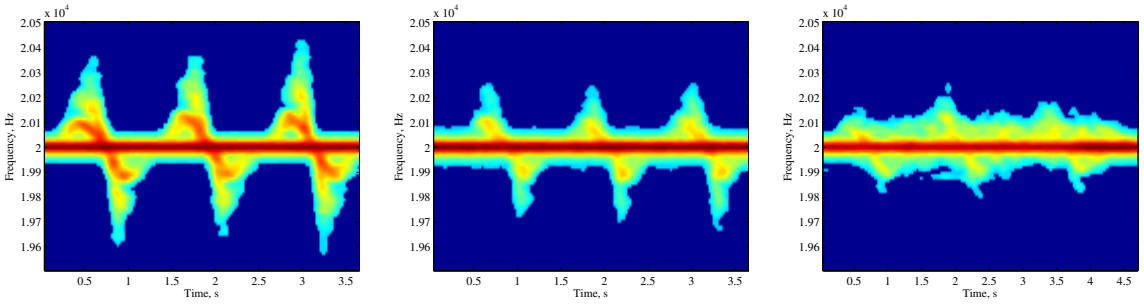


Figure 3.8.: Three different scenarios are depicted here. In (a), the smartphone is placed stationary on the table. In (b), the phone is hold in one hand. In (c), the phone is hold in one hand while walking through a corridor. The other hand is performing a wave gesture towards the sensing devices causing the repetitive movement in the Doppler broadening.

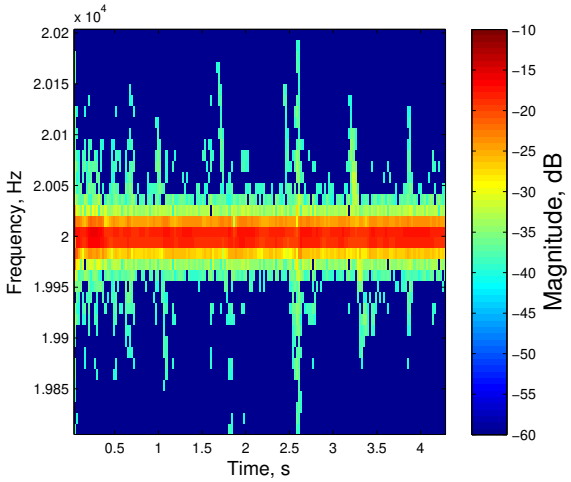


Figure 3.9.: Spectrogram of gestures performed on a mobile device put inside a trouser pocket. The Doppler shifts have a very low magnitude but can be recognized. This is due to the attenuation of the sound wave travelling forth and back through the trouser material.

woods or fields, via the reflected signal. Results however show that a clear signal cannot be extracted due to the weak echo and high 'noise movement', as discussed earlier. Furthermore, commercial products to strap the device on the runner's arm often cover either the transmitter or the receiver, which weakens the outgoing or incoming signal. Here, noise estimation in the proximity of certain entities, as thoroughly researched in [SKJS15], might be applied to clean the incoming signal as well as applying calibration.

3.1.4. Technical challenges

Our experiments disclose that using the Doppler frequency shift is a feasible technique to recognize gestures as well as activities and general movements in the proximity of the device. In a constraint setup, e.g. a stationary deployment of the mobile phone, the received signal is adequate to detect a variety of gestures. Especially two-handed ones, where the motion of each hand is diverse, are a domain where our approach has its advantages. On the other hand, real-life applications require noise handling depending on the circumstances. Yet the technique is still usable if the phone movement is not too excessive. Non-stationary applications on the other hand induce greater challenges in signal processing. The main limitation is due to strong noise via movement or from user's environment. As long as the mobile device is placed inside the clothing, the signal strength is strongly attenuated through the clothing material. Finally, additional noise from user's own movements, like in case of the application on a jogger's arm, hinders the detection of additional Doppler shifts from the environment.

Ambiguous Doppler Reflections In addition to noise handling, certain scenarios may require the algorithm to consider and handle multiple reflections of the emitted peak signal. To illustrate this effect to a higher degree, we repeated the wave gesture using a large sheet of paper. In this scenario, the original signal, emitted by the mobile phone, gets reflected once upon reaching the paper. This is the signal we receive via the device and it exhibits a Doppler shift. However, it is reflected once more by the table, making it seem like a weaker version of the original peak signal. Again this signal is subject to a Doppler frequency shift upon reaching the paper making it look like there is another Doppler shift with double the velocity of the first. The effects of this phenomena as shown in Figure 3.10 can be observed more intensely when using large reflective surfaces. In a setup using hand gestures it is more commonly witnessed as a second flare being present at times when the hand is in close proximity of the phone (cf. the visible flares on every second zero-crossing in Figure3.8 (a)). For known movements, this issue can be avoided by excluding multiples of the first frequency shift. However, if there is in fact a second movement twice as fast, things can get ambiguous and hinder reliable gesture recognition.

Missing Directivity Newer smartphones include two or even three microphones. It is difficult to determine the exact position of each microphone for each device and address the right one. This issue also applies to the position of the loudspeaker, leading to a lack of generalizability between multiple devices. Bannis [BZP14] showed that by using a setup with more than one receiver or transmitter, one could yield the Doppler directivity by combining the collected measurements.

Quality of Hardware Another negative aspect concerning a smartphone's hardware is its quality. Usually, the modules for microphone and speaker have to be as small as possible for design purposes, which restricts audio quality in both the listening and recording domain. To compensate for this, a possible algorithm must be robust to noisy data, especially since data from one phone might be different in magnitude than from another even when performing the same activity or gesture. Occasionally, speaker and microphone are co-located, leading to unpredictable amplitude variations in the recording, as pointed out by Rajalakshmi [NGW15]. For experiments that require a stable amplitude, these kind of phones, e.g. the Galaxy Nexus, cannot be utilized. In addition,

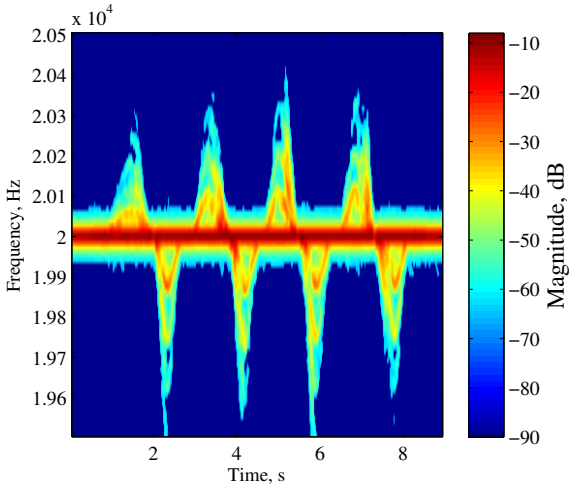


Figure 3.10.: Received spectrogram showing multiple reflections of the original Doppler shift. For instance at 6 seconds, there are three distinct doppler shifts visible. A fourth can be adumbrated. The maximum extend of the first shift at this timestep is just lower that 1.99 x10^4 Hz. Every follow up shift (around 1.975 x10^4 Hz and 1.96 x10^4 Hz) occurs due to multiple reflections of the sent signal.

models with automatic noise cancellation are not suitable for the designed application either, as they cancel out the carrier frequency and the Doppler patterns around this frequency. Finally, using the phone's speakers at maximum output increases the amount of eigen-frequency excitation, which may fall into the human range of audibility and disturb the user.

3.1.5. Useful findings and conclusions

According to the conducted exploratory study, we illustrated the feasibility of a smartphone using its native microphone and speaker to recognize nearby movements. We presented a number of experiments using ultrasound Doppler frequency shifts to detect gestures, general motion and activities of a user. Various advantages of ultrasonic sensing can be seen such as simultaneous detecting several opposite movements, detection of fine-grade body movement by leveraging the micro-Doppler information and robustness against illumination changes. We showed that our approach can be successfully applied in stationary setups or when holding the phone. When the phone is worn on the body, overlying clothes may strongly attenuate the received signal. Although we have not tested our approach on a wide range of target platforms and with many users, our results represent a starting point for other application practitioners. As we are filtering the signal below 18 kHz and only applying the silhouette of the Doppler broadening information, the spectrum of natural speech is removed and thus the issue of privacy, while collecting acoustic signal, is resolved.

In the course of our later researches, we will further investigate machine learning approaches for more complex activity recognition. We will also target the challenge of data diversity with different hardware. Using multiple devices and microphones, we could overcome limitations in directivity and noise rejection. But multi devices setup is impracticable and cannot be explored beyond the lab setting. Therefore, we restrict our research to single device application. As stationary setups provide promising results, the following application mainly focuses on using stationary placement for ubiquitous sensing.

Figure 3.11.: The three different sport activities: *bicycle, squat*, and *toe touch* exercise are shown from left to right. The placement of the smartphone is close to the wall to ensure a stronger back-reflection by leveraging the multi-path reflection of the Doppler signal.

3.1.6. Study 1: Design mobile application for selected activity recognition with commercial smartphone

This section is based mainly on our work published in [FKK*18]. Using the same smartphone application introduced in the previous section, we aim at recognizing up to three whole-body sport exercises, including *bicycle, squats* and *toe touch* exercises. The type of these exercises is depicted in Figure 3.11. These three exercises are fairly simple, but diverse, such allowing us to achieve a relatively high recognition accuracy. Our development database is acquired from 14 individual participants. The structure of our approach in this use-case are grouped into following aspects:

1. Investigation of hardware limitation and placement
2. Build a database of 14 test participants to showcase the proof-of-concept
3. Investigation of various classification schemes based on inductive biases by leveraging feature engineering with domain "expert" knowledge as prior
4. Investigation of automatic feature extraction by using 2D-CNN architecture for the end-to-end learning architecture based solely on the data

3.1.6.1. Device Placement

In our previous work [FKGP*15], we have shown, that back reflection of a mobile device placed inside a trouser pocket is quite weak and noisy. Therefore, we focus on remote sensing of the targeted activities by placing the device not directly on the body. Empirical test study shows the best placement of the sensing device in order to obtain a strong back scattered signal is close to the wall or heading towards the corner of the wall. From both locations we get enough reflections back to the device, which can be observed from the amplitude of the periodical frequency modulation around the center carrier frequency.

Though a large ground plane reflects well the echo back to the sensing device laying above, the corner of a wall leads to the strongest back reflection of echoed signals. This is due to the multipath reflection additive from the corner of the wall. This effect of corner reflection has already been proven useful in radar applications [SYY*13]. A corner reflector is often used to calibrate the radar, since it has a very high radar-cross-section (RSC) and

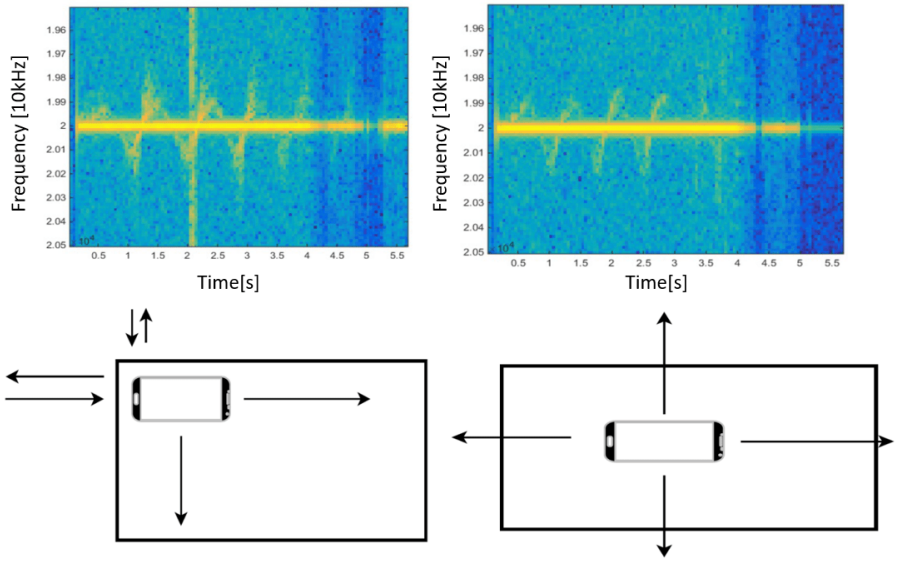


Figure 3.12.: Bottom left shows the placement of the device on the corner of the table directly facing a wall corner and its spectrogram is shown in the top left panel. Bottom Right: The device is placed in the middle of the table with no line-of-sight directly to the corner and its corresponding spectrogram is shown in the top right panel. The back reflection of the corner placement is stronger due to the multipath reflection.

guarantees for strong back-scattering even with a small size and the high RCS is maintained over a wide incidence angle. Therefore, in our experiment, we collected the sport exercises data samples in the close vicinity of a wall as depicted in Figure 3.11. We show this effect on a small experiment by placing the smartphone on a large desk and recording simple waving gestures above the sensing device. We tested two different placing positions, which are in the middle of the table and on the edge of the table facing a corner of the wall. In the first case by placing the device in the middle of the table, no direct line-of-sight to the corner is given. In the second case by placing the device on the edge of the table directly facing a wall corner, we simulate the setup of a corner reflector. It can be clearly observed from Figure 3.12 that the placement on the corner close to the wall has got the strongest back reflection.

3.1.6.2. Classification Methods and Evaluation

We conducted a small test study to collect exercise data in order to test the different classification methods for our targeted application. We invited 14 participants ranging from 25 - 28 years old to perform the three given sport activities. The 14 participants are grouped in 12 males and 2 females, with a weight ranging from 60-80 kg for males and 50-55 kg for females and a body height ranging from 150-180 cm and 155-160 cm respectively. We asked each participant to perform each activity fifteen times, which makes in total 45 samples each. The device we used to collect the activity data is a Google Nexus 5X. To condition the measurement setup, all the exercises are performed in the same location. The exact placement can be seen from Figure 3.11.

Exercise	Minimum (TS)	Maximum (TS)	Minimum Duration[s]	Maximum Duration[s]
Bicycle	13	31	0.60	1.44
Squats	12	42	0.55	1.95
Toe touches	11	25	0.51	1.16

Table 3.1.: The table shows the time duration for the fastest and slowest speed of each sport activity performed by the test participants. The abbreviation TS stands for time window segment.

For the data acquisition process, we implemented a mobile application to remotely annotate and collect the exercise data by an external manual instructor. The same application is installed on two smartphones functioning as a master and a slave device. This module can discover the peers registered under the same WiFi and connect them together. One device is used as a slave device to collect the data while users are performing sports activities close to it. The second device is used as a master device to remotely and manually label the activities performed by another instructor. This setup enables the user to practice unobtrusively without interruption. As for installation, there is no need for any external hardware. The data collection app has to be installed and running on the smartphone. It then can be placed on the ground and start to collect the echo of performed activities in the vicinity of the device. Since the backscattered signal is strongly dependent on the signal power emitted by the device itself, we have turned the device volume to the maximum level to assure enough signal back propagated by the reflection of the body. The signal strength overcomes the attenuation of the two-way-paths.

For testing various classification schemes, we evaluated the data from the test study in a post-processing fashion using Python and the scikit-learn library [PVG*11] on a normal Desktop PC. The processing steps aim at enhancing the signal quality in preparation for classification. First, we convert the received time signal to the frequency time domain by leveraging the STFT technique. For each time slice of 93 ms, corresponding to 4096 time samples with a sample frequency of of 44.1 kHz, a $N = 4096$ FFT is applied to the sliced time window to extract the frequency distribution for this time window. With an overlap of 50 %, the actual time resolution of each sample point in the frequency time domain corresponds to a observation time window of 46.5 ms. The spectrogram X represents the magnitude of the complex signal Z_{STFT} by applying the equation $X = |Z_{STFT}|$. In the pre-processing step, we performed signal normalization and the center frequency removal. The filtering is done in the frequency domain by applying a segmentation filter to only extract the frequency component ranging from 19 kHz to 21 kHz. The normalization is done such that the spectral amplitude lays within 0 and 1 by applying Equation (3.6).

$$X_{norm} = \frac{X - X_{min}}{X_{max} - X_{min}} \tag{3.6}$$

We further cancel out the central carrier frequency by applying the Equation (3.7):

$$X_{STFT,k} = |X_{STFT,k} - X_{STFT,k-1}|^2 \tag{3.7}$$

and then square the remaining changes. The resulting spectrogram can be viewed in Figure 3.13.

The processed spectrum for the three targeted exercises are depicted in Figure 3.14. The exercises *bicycle, squat*, and *toe touches* are depicted in the order from left to right. This normalized Fourier spectrum over time served as basis features for the classifiers. Derived from the general statistics of the collected data, we then conditioned the common time duration of these three different activities. From Table 3.1, we can see the time duration for the fastest and slowest speed of the performed activities.

We chose the average time performing each activity and thus have to inter- or extrapolate the collected data samples to make them into comparable data formats to feed them to common classifiers. A simple linear interpolation scheme is used to condition the data. Therefore the input data has the dimension of 186x40 ranging from

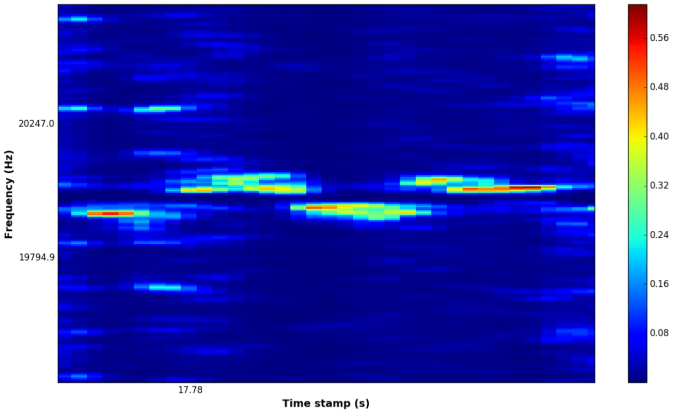
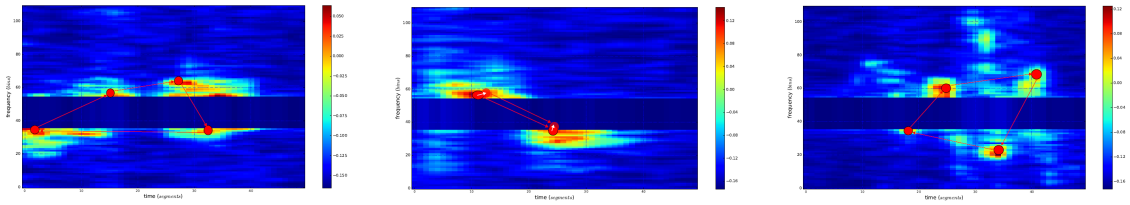


Figure 3.13.: Spectrogram of a normalized and carrier-free input signal.



(a) (b) (c)

Figure 3.14.: The processed spectrum for the three targeted exercises are depicted here. In (a), the spectrum for the bicycle exercise is shown. In (b), the spectrum for the squat exercise is shown. In (c), the spectrum for the toe touches exercise is shown.

Estimators	Max Features	Tree Depth
300	sqrt	100

Table 3.2.: Hyperparameters for Radnom Forest as a multi-Label classifier.

Kernel	γ	Penalty parameter
Linear	0.0001	1

Table 3.3.: Hyperparameters tuning for SVM as a multi-Label classifier.

frequency to time domain. 186 frequency bins represent the frequency range of 19 kHz to 21 kHz and 40 time samples represent a duration of 1.85 s. For the current evaluation, the start and end markers are labeled manually. For later online application, the sliding window approach should be used to classify different sports exercises.

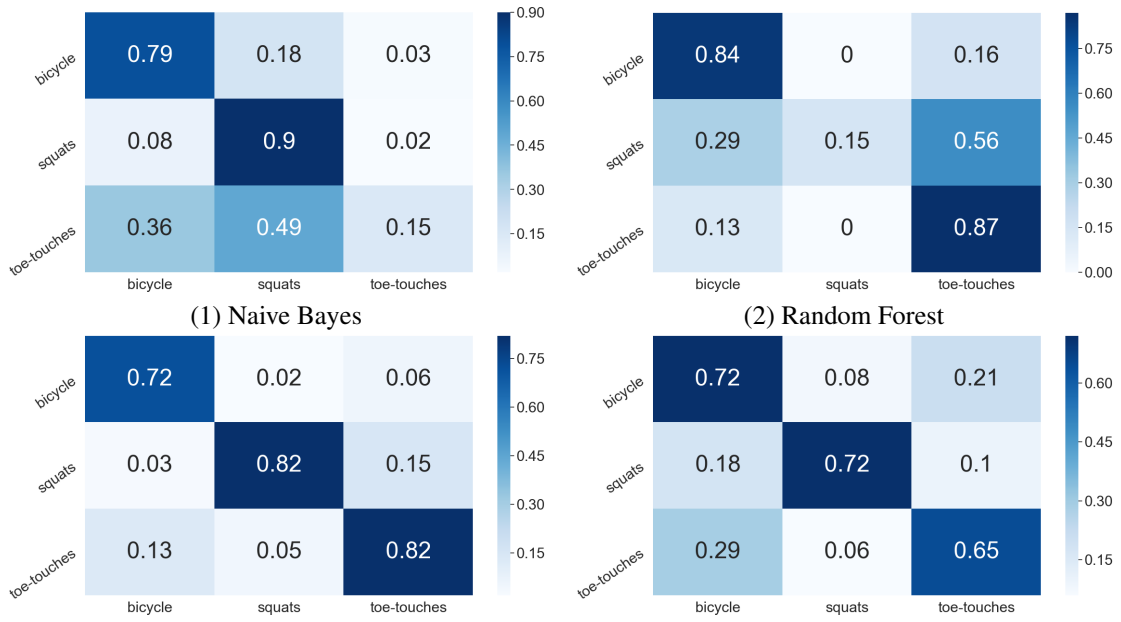
We investigated different classification schemes such as Naive Bayes (NB), support vector machines (SVM), random forest (RF) and AdaBoost to evaluate the classification results. To perform the classification, we further divide all collected samples into 80 % training samples and 20 % test samples. This excludes the test samples during the training phase. In this way, we can generalize the outcome of our classifier on the unknown test inputs. We further applied the 10-fold cross-validation to show the macro-precision and the macro-recall to the different classification schemes. In the following section, we describe the evaluation results for all the different classifiers evaluated and try to offer the best practice classifier for our proposed application.

3.1.6.3. Evaluation Results

The confusion matrix for different classifiers are given in Figure 3.15 using the same split on training and test data. Naive Bayes classifier has a very high misclassification rate for the class toe touches compared to the class bicycle and squats. It often got the class toe touches confused with the other two classes. The class squat has the highest classification accuracy of 90 %, while the class bicycle follows with an accuracy of 79 %. Although the confusion matrix for Random Forest shows a relatively high accuracy of 84 % for the class of bicycle and 87 % for the toe touch class, but it still works poorly on the class of squat. The hyperparameters for Random Forest are shown in Table 3.2. We use 300 different single estimators with the maximum tree depth of 100 to setup our classifiers.

The confusion matrix for Multi-classes SVM using the one-versus-rest classification show however clearly better results. In our case, we used a linear kernel and a regularization parameter to prevent our classifier from overfitting. The hyperparameters to setup the support vector machine can be seen in Table 3.3. In this scenario, the misclassification problems between squats versus bicycle or toe-touches or the misclassification between toe touch versus squats or bicycle have been drastically reduced. We obtain an accuracy of 72 % for bicycles, 82 % for toe-touches and 82 % for squats. The overall performance is more stable and the results are much better than the Naive Bayes classifier or the Random Forest. Despite the slightly higher accuracy for the class of toe touch in the Random Forest classifier, we still cannot expect the RF classifier to outperform the support vector machine.

For AdaBoost classifiers we used 300 weak estimators to setup the hyper classifier. AdaBoost proved to be the better classifier compared to Random forest, as the confusion matrix is more balanced. The main diagonal of the confusion matrix shows relatively good results for all the three classes. The same accuracy of 72 % was achieved for the classes squats and bicycle and an accuracy of 65 % was achieved for the class toe touches. From all classifiers observed so far, the AdaBoost shows more superior performance than Random Forest and Naive



(1) Naive Bayes (2) Random Forest

(3) Support Vector Machine (4) Adaboost

Figure 3.15.: The confusion matrix for different classifiers are depicted. (1) shows the confusion matrix for the Naive Bayes Classifier. (2) shows the confusion matrix for the Random Forest classifier. (3) shows the confusion matrix for the Support Vector Machine classifier. (4) shows the confusion matrix for the AdaBoost classifier. As can be clearly seen from the results, AdaBoost classifier performs better than Naive Bayes and Random Forest. Though with higher computational load, it has a worse accuracy than support vector machine. The support vector machine shows the best recognition results with fewer misclassification rate.

Bayes. But compared to the Support Vector Machine, this classifier showed worse results with a considerably higher computational load.

To further investigate the classifiers performance, other statistic measures such as the ROC curve and the AUC area are considered. It is clearly shown that the Naive Bayes classifier is a really bad classifier for this kind of data as the classifier is only slightly better than random decisions. The ROC curve for a binary random decision classifier performs as a straight line through the origin with an inclination of 45 degree. The ROC curve for the Random Forest classifier can be seen in Figure 3.16 (2). The curves of the RF classifier is much curved towards better performance compared to the curves of the NB classifier, which again indicates that the Naive Bayes classifier works poorly. The same behaves for the area under curve, which intends that the RF is a better suited classifier than Naive Bayes for our data set. The performance of toe touch class and the bicycle class are quite similar and better than the class squats, as the area under curve is larger. The ROC curve for the Support Vector Machine shows the best result over all classifiers investigated so far. The performance of all the three classes are quite similar, as the area under curve are nearly identical. This is a good sign for choosing Support Vector Machine as a robust classifier for our application. The ROC curve for the AdaBoost classifier can be seen in Figure 3.16(4) as well. The class bicycle performs the best, where the class toe touch and the class squats

	Naive Bayes	Random Forest	Support Vector Machine	AdaBoost
Precision	64 % ± 0.38 %	77 % ± 0.22,%	**84** % ± 0.33 %	77 % ± 0.33 %
Recall	61 % ± 0.17 %	68 % ± 0.16 %	**83** % ± 0.35 %	75 % ± 0.43 %

Table 3.4.: The macro-precision and macro-recall for the 10-fold cross-validation is presented here.

	Naive Bayes	Random Forest	Support Vector Machine	AdaBoost
Accuracy	54.91 % ± 3.1 %	64.20 % ± 2.3 %	**74.18** % ± 2.0 %	64.83 % ± 2.1 %

Table 3.5.: The accuracy using leave one subject out cross-validation is presented here.

behave more similar. These classifiers are steeper than in the case for the Random Forest, but not equally good as in the case of the support vector machine.

In order to give a more generalized score for the given classifiers, we provide the macro precision score and the macro recall score for the 10-fold cross-validation case in Table 3.4. The macro-precision and the macro-recall are the average precision and recall for all the 10-fold cross-validation. The accuracy of the different classification results using leave one subject out cross-validation is depicted in the Table 3.5. They provided the same findings as discussed so far.

3.1.6.4. Activity recognition using convolutional neural networks

Activity recognition with neural networks is, however, an upcoming field that has recently emerged [RC16]. Various deep learning neural networks have been tested on time series data for human activity recognition. The CNN is a fairly simple network structure. They are made up of neurons that have learn-able weights and biases. The difference against fully connected network architecture is however that it only takes into account for the spatial information from neighbouring nodes, which means that CNN captures local dependencies and has moderate capacity opposed to fully connected neural network architectures with the same amount of layer nodes. This local connectivity is of vital importance, since for activity recognition the temporal actions are correlated. Another desirable property of CNN is its scale invariance by leveraging the pooling layer. It can detect the correct features in different scale levels similar to a pyramid structure. This effect corresponds to human activities performed at different speeds.

The model architecture of the convolutional neural network used here is depicted in Figure 3.17. Lower layers extract simple low-level base features such as edges or curves, whereas higher layers extract more abstract high level features out of combinations of base features. The main advantage of using such end-to-end network is its ability of automatically extracting useful features or correlated structures based on the training data only.

Input to the neural network is the 2D spectrogram dataset of the dimension 186x40 and the output from the softmax layer are 3 nodes corresponding to the probability to each of the three classes. The higher the output probability is, the more probable is the sample belonging to this class. The output of each of the softmax node can be expressed by the Equation 3.8.

$$softmax(z_i) = \frac{exp(z_i)}{\sum_j exp(z_j)} \tag{3.8}$$

The italic character i represents one of the three possible classes and $j = 1..$#classes. The sum of the softmax output results in 1, thus each element results to a probability value of classifying this sample to this certain class. In the lower network layer we use four convolutional layers to extract the local dependencies in each layer. The

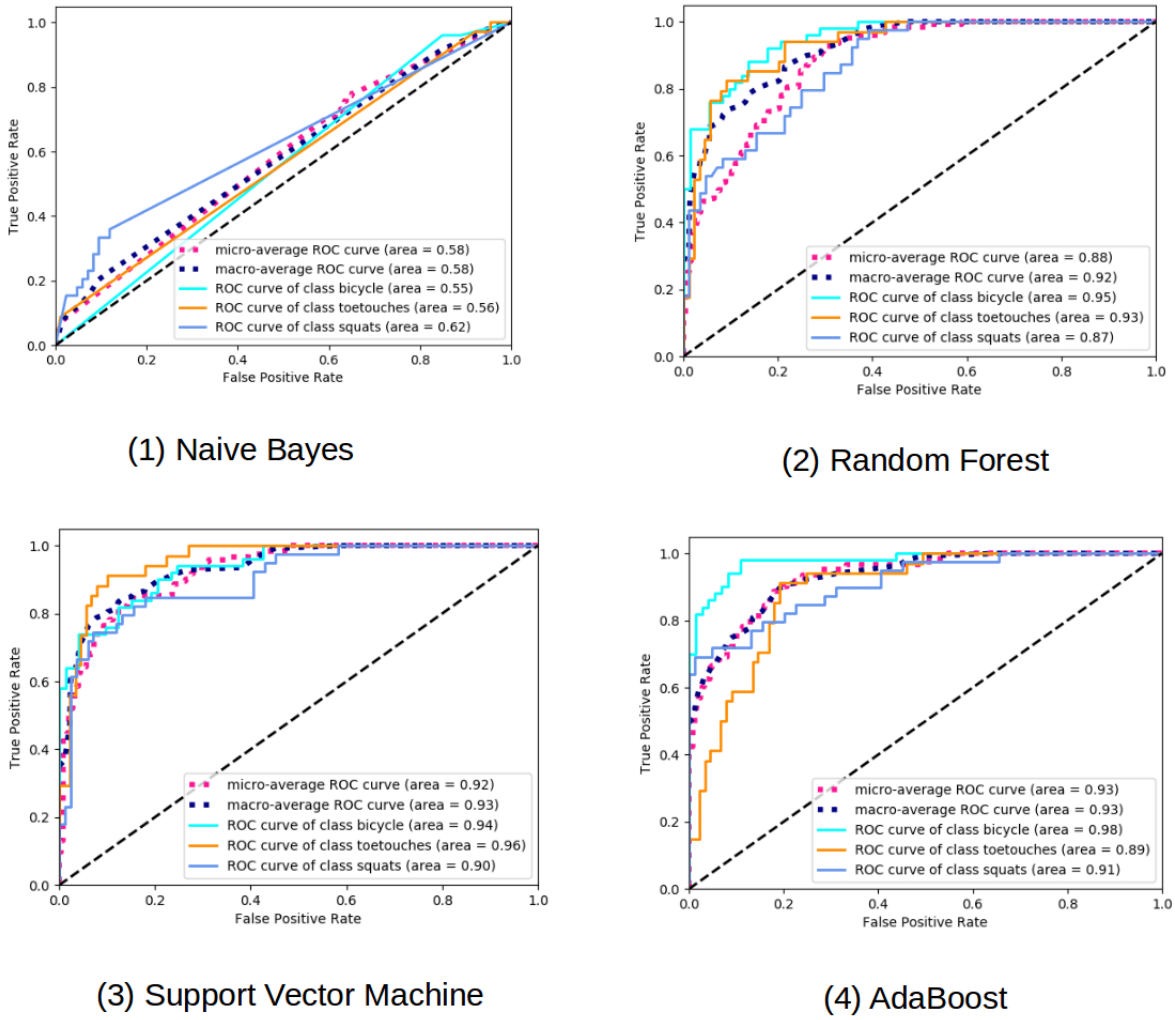


Figure 3.16.: The ROC curve for different classifiers are depicted. Top left shows the ROC curve for Naive Bayes Classifier. Top Right shows the ROC curve for the Random Forest classifier. Bottom left shows the ROC curve for the Support Vector Machine classifier. Bottom right shows the ROC curve for the Adaboost classifier. As can be extracted from the results, the Naive Bayes classifier performs the worst, which is only slightly better than random guessing. The Support Vector machine performs the best compared to the other classifiers. For all three classes, the ROC curves behaves similarly and shows the most robust results.

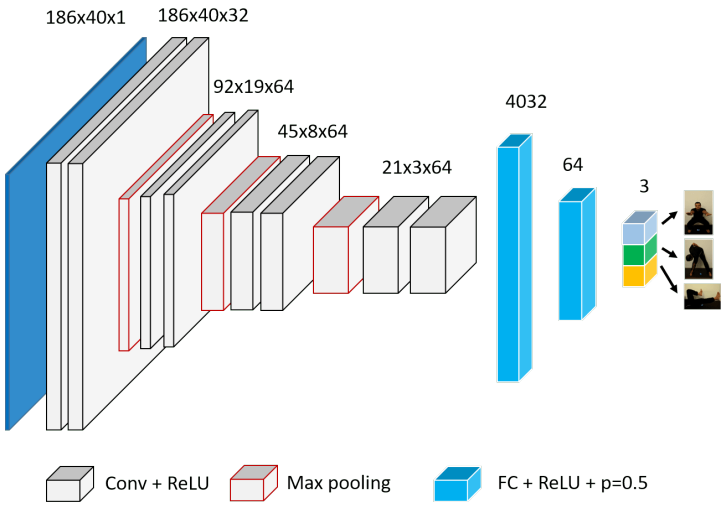


Figure 3.17.: The CNN architecture applied for our recognition task targeting three sports activities is illustrated in this figure. Four successive convolutional layers are used to automatically extract local features. Rectified linear unit (ReLU) activation function is used following each convolutional layer. Max pooling layer is used to successively reduce the size of the input sample. Two fully connected layers (FC) are used to combine the local descriptors from the final convolutional layer. A dropout layer with the dropout rate of 0.5 is used after the second FC layer to reduce overfitting. The softmax layer is used to calculate the final classification probability.

	Predicted		
True	bicycle	squats	toe-touches
bicycle	**.88**	.06	.06
squats	.03	**.91**	.06
toe-touches	0.	.03	**.97**

Table 3.6.: Confusion matrix of the convolutional neural networks is depicted.

size of the kernel filters connects only part of the input nodes to draw the local dependencies. The rectified linear units (ReLU) is used as activation function following each convolutional layer. There are 3 successive max pooling layers used to reduce the input dimension size and enable the convolutional layer to extract features from three different scales. This step corresponds to the different user speed. After each max pooling layer, the size will be reduced to half of the previous size. It is like a pyramid-based learning that you are able to get features from different scale levels. At higher network layer, two fully connected layers are used to connect all the learned local descriptors from the final convolutional layer. A dropout layer is used as a regularization term and helps the network to avoid over-fitting. The ratio of the dropout layer followed the fully connected layer is set to 0.5. The softmax layer returns the prediction of the final classification. The weights in each individual layer are learned and updated using RMSprop as Optimizer and ReLu Layer as activation layer. The equation for ReLu activation layer is introduced by Hinton [NH10]. In RMSprop, the gradient descent learns the weight at activation range only. The equation for RMSProp is first proposed in [TH12]. The learning rate for each weight is updated by dividing a running average of the magnitudes of recent gradients for that weight. The advantage is that it reduces the problem of overshooting from the global optimum and makes the network quickly converges to the global optimum.

The confusion matrix for CNN applied to our collected exercise data is given in Table 3.6. The correct classification accuracy for all three classes outperforms the support vector machine, indicating its superiority in extracting local features to discriminate these exercises. The misclassification rate are relatively low compared to the other classical machine learning approaches. The Leave one subject out cross-validation for CNN has an average accuracy of 98.31 % and the average accuracy for the 10-fold cross-validation is about 98.36 %.

3.1.6.5. Discussion and conclusion

Here, we propose a stationary setup of using a commercial smartphone to detect three whole-body activities with Doppler sensing. By leveraging only the built-in hardware components of an unmodified consumer-grade smartphone, we are able to recognize the three sport exercises, such as *bicycle, squats* and *toe touches* with a fairly well performance. We collected data from 14 different participants and evaluated the data on various classification methods including Naive Bayes, Support Vector Machine, Random Forest, AdaBoost and Convolutional Neural Network. The evaluation results are done in post-processing mode using Python with the scikit-learn package and the Tensorflow library for deep learning. The start and end markers are manually labeled and should be extended to use sliding window approach in the later online application.

The choice of using an off-the-shelf unmodified smartphone to track sports activities is mostly due to its flexibility against wearable sensing devices or other stationary sensing technologies which require extra installations to the physical environment. Compared to visual input such as Kinect, it further preserves the privacy of the user by avoiding visual input. The pre-processing step is kept simple to test the generalized performance for all the classifiers despite the classical machine learning approaches like SVM or neural networks. The SVM showed the most robust and best performance compared to the other conventional machine learning approaches with respect

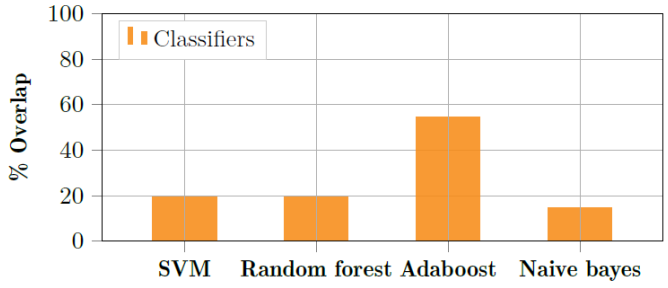


Figure 3.18.: Overlap precision comparison from 0% to 100% on trained classifiers. This study aims to investigate the robustness and performance of signal coverage with respect to trained classifier for real application.

to our target application. AdaBoost from the ensemble learning families also provides better performance compared to Random Forest and Naive Bayes. In comparison to SVM, AdaBoost requires much more computational effort due to the ensemble method. CNN with its inherent ability of automatic local feature extraction especially for 2D image data, outperforms the traditional classifiers in the final accuracy. An improvement of accuracy can be observed in all three classes.

The current labeling process is done for each exercise repetition, ensuring the 100 % coverage of the underlying exercise. To simulate sliding window approach for real-time application, we further investigated the offset shift conducted for one randomly selected participant. We reduce the coverage from 100 % to 0 % and investigated the robustness and performance of the trained classifiers with regard to the signal coverage. The result is depicted in Figure 3.18. The SVM and Random Forest classify correctly classify for a 80 % signal coverage, while Adaboost still works for a coverage of 45 %. Naive Bayes classifier requires at least a coverage of 85 %.

However, it should be noted that all data collected in this work has been created using a Google Nexus 5X. The quality of the built-in hardware component and the placement of the built-in microphone and loudspeaker can lead to deviating classification result. Since the frequency range of speech is not considered, environmental sound noise is not a problem for the processing algorithm. However, in order to ensure best back-scattering of useful event signals, the sensing device has a preferred placement. In the following work, the aim is to reduce this constraint of sensor placement and thus enabling us to practice anywhere at anytime.

3.1.7. Study 2: Enable more complex and realistic sport activity recognition with less restrictions

The goal of our previous application scenario is restricted to only recognise three completely different exercises. We thoroughly investigated the placement in order to receive the strongest back reflection. However, this introduced further limitations and constraints to the developed application. To intensify the back reflection, the device was positioned on a smooth surface close to the wall. In order to loosen these constraints, such as sensor placement, more efforts have to be taken for data processing. Mitigating the position restraint is a prerequisite in order to practice everywhere at anytime. In this proposed use-case, I further focused on a more realistic set of activities. Compared to the previously targeted three whole-body exercises, now we look at more similar and complex exercises. This makes the recognition of whole-body workout exercises more challenging in its generalization task. This section is primarily based on our work published in [FKK20b].

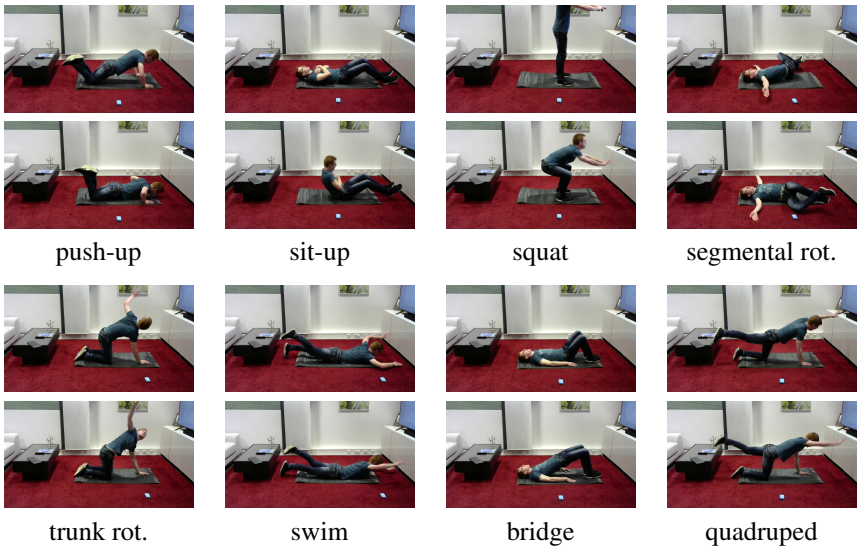


Figure 3.19.: Figure shows the eight workout exercises we collected in our living laboratory. Figure also illustrates the position of our sensing mobile device with respect to the performing user body.

In this section, we focus on remote sensing with a commercial smartphone device to recognise eight workout exercises. This set of activities are considered to be more realistic and less constructed as in the previous setup. A modern smartphone possesses the computing power comparable to a working station in the mid 80s. Equipped with all the integrated hardware sensors, researchers are able to develop lots of interesting applications. By leveraging the integrated smartphone loudspeaker to emit a continuous signal of 20 kHz, we actively turn the device into an active sonar sensor. This approach is proved to be feasible in the previous sections. By analyzing the echo signals and extracting features from the transformed frequency time spectrum, we are able to train several carefully designed end-to-end learning classifiers. The sequence model, the fine-tune model, and as well as the 2D CNN structures are investigated. Finally, we examined the few-shot method to further increase the generalization ability of the performing model.

The eight different workout activities: *push-up, sit-up, squat, segmental rotation, trunk rotation, swim, bridge*, and *quadruped* are illustrated in Figure 3.19. The contributions of our work are concluded in the following aspects:

- Investigate more sophisticated workout exercise by leveraging micro Doppler profile from spurious human body motion.
- Loosen the constraint of sensor placement by paying more effort in the data processing and segmentation stage.
- Count the exercise repetition with peak detection algorithm on pre-processed Doppler spectrum

The overall structure of this section is as follows: we shortly explain the physical sensing principle with improved data processing method of our proposed application. The detailed processing pipeline of this work is also provided. Then we propose three different end-to-end learning architectures and justify our design choices. To further improve the generalization ability on the small data amount, we examine one approach from the few-shot classification learning. The setup for the final evaluation is discussed in detail, where we first introduce the

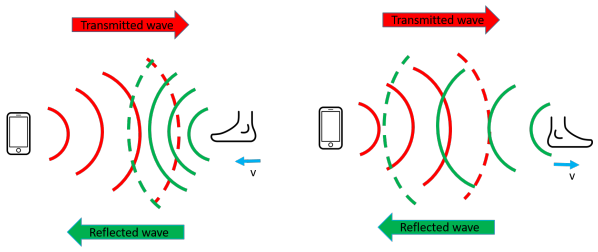


Figure 3.20.: Figure illustrates the physical working principle of Doppler sensing. When the measuring object is moving towards the receiver, the distance of the wave-front decreases and thus leading to a positive frequency shift. Opposed direction leads to increase in the wavelength and thus indicates a negative frequency shift.

statistics of the test population, followed by a description of the experimental setups and the discussion on the results. A conceptual method of the repetition counting is also proposed. With regard to the design architectures and the conducted experiments, we identify the challenges and provide viable solutions.

3.1.7.1. Sensing Theory

The sensing method we are using is called Doppler sensing. We use a commercial smartphone, (Samsung Galaxy A6 2018) and turned the device into an active sonar system by emitting a continuous sound wave with a carrier frequency of 20 kHz. Other specifications are already introduced in the previous sections. According to the physical definition, we can relate motion and speed to frequency by measuring the Doppler shift in the frequency domain. Doppler shift corresponds directly to the relative speed to the receiver. A relative motion moving towards the receiver is causing a positive Doppler shift, while a relative motion away from the receiver is causing a negative Doppler shift. The working principle of the Doppler broadening caused by motion can be seen in Figure 3.20. Doppler motion is adding relative changes to the center frequency. The additive speed in both directions is causing the operating wave-front to change its form to either expanding or contraction.

To better resolve this Doppler shift, we first have to transform the time signal into frequency domain by applying fast Fourier transformation. The number of Fourier coefficients is directly related to the resolution of the frequency domain. However, according to the frequency time uncertainty [DD15] we cannot have both fine resolution in time and frequency. Here we apply the trick of zero padding to achieve better frequency resolution even with a small time window to guarantee a fine-grained time resolution. The working principle is illustrated on a simple sinus wave in time as depicted in Figure 3.21. By adding additional zeros to the time series window, we are synthetically generating finer resolution in frequency domain without adding more information to the time domain. This smoothness and fine resolution in the frequency domain allows us to better detect the Doppler shift caused by relatively slow body motions.

A discretely sampled input wave file received by the device internal microphone is representing the time series encoded with the repetitive motion patterns from the workout exercises performed in the vicinity of the sensing device. For each time series, a frequency time spectrum can be calculated to reveal the Doppler profile over time. The resolution in time and frequency can help the classifier to better model the data. Since the motion speed for workout exercises are fairly slow compared to hand gestures, the corresponding Doppler shift in frequency is thus minor as well. If the resolution in frequency is coarse, the Doppler broadening is easily overshadowed by the large center frequency signal. The frequency resolution is inverse proportional to the time resolution. Therefore, in order to have a high frequency resolution, a large observation time window is required. This leads to a coarse

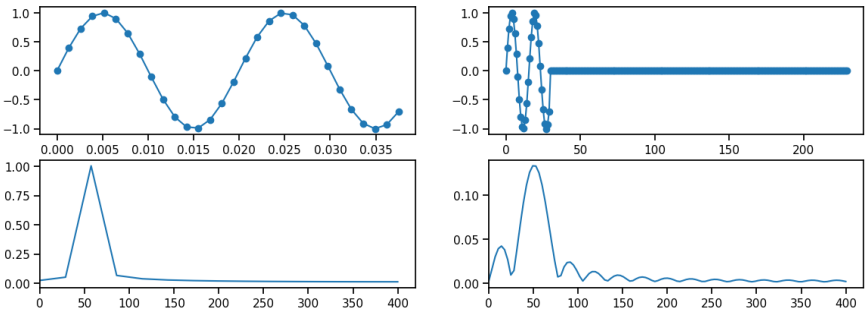


Figure 3.21.: Figure shows the frequency resolution with and without zero padding. By adding additional zeros to the time series window, we are synthetically generating more fine resolution in frequency domain without adding more information to time domain.

Term	Meaning	Values
f_s	sample frequency	44.1 kHz
Δt	time resolution	46.5 ms
f_0	carrier frequency	20 kHz
v_0	speed of sound wave	340 m/s
N_{FFT}	number of FFT points	12288
Δf	frequency resolution	3.6 Hz
Δv	speed resolution	3 cm/s

Table 3.7.: The table shows the parameters of the time-frequency representation of out proposed system.

time resolution and a large response time of our application which makes it difficult to build a system with nearly real-time feedback. Now, by applying zero padding to the time signal, we are able to selected relative small time windows but still having relatively smooth frequency resolution.

Now, we want to list the numbers to provide you a feeling of the orders of magnitude we are talking about. The audio sample frequency of the smartphone is 44.1 kHz. For each 4096 time samples, a fast Fourier transformation is calculated. With an overlap of 50 %, we achieve a time resolution of 46.5 ms. We use the zero padding for the entire time window to have 12288 values, which corresponds to a frequency resolution of 3.6 Hz of each frequency bin. This results in a relative speed resolution of 3 cm/s, thus enables us to have a three-folds finer resolution even for a slow motion speed. An overview of these numbers can be seen in Table 3.7.

The device specific characteristic is provided by the plot showing the impact of user-phone distance to the performance. In Figure 3.22 the signal profile shows the attenuation caused by the distance which has a strong impact on the performance of the classifiers. The larger the distance, the weaker the reflected signal due to 2-way propagation. The sensing device is placed at a height of 81 cm, while a participant is approaching the device from a distance of 250 cm with a step size of 50 cm. Due to the height of the device placement, the dominant component in the echo results from the reflection of the abdomen. The first positive Doppler at 6 s is caused by taking the first step from 250 cm to 200 cm from the sensing device. An increase in signal strength is clearly visible, indicating a better signal performance for a shorter distance. Thus, we select the distance of 50 cm to achieve a good trade-off between remote detection and strong enough back reflection to perform further

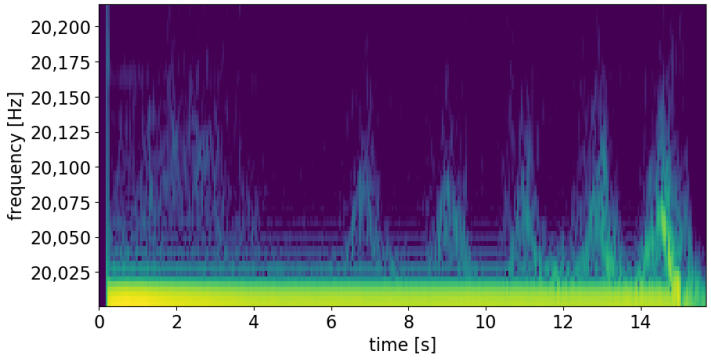


Figure 3.22.: The impact of user distance to the sensing device is illustrated here according to the signal strength in the echo signal. The device is placed at a distance of 250 cm away from the user, while the user is slowly approaching the device with a step size of 50 cm.

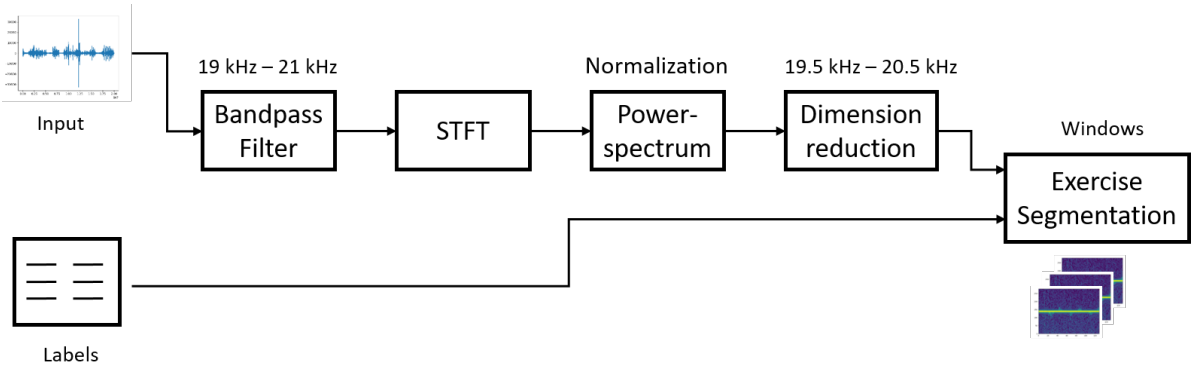


Figure 3.23.: Figure shows the processing pipeline starting from the raw audio input to the segmentation step.

processing. The distance of the device should not restrict participant from performing actions as naturally as possible, while still ensuring the weakest signal to be detectable by the sensing device.

3.1.7.2. Data Processing

In this section, we will introduce a series of processing steps to prepare our data for the classification networks. For the data acquisition task, we developed an android application to get the exercise data and its corresponding labels. The processing pipeline is illustrated in Figure 3.23.

A butterworth bandpass filter in the 6^{th} order is applied on the raw input signal to filter out natural speech and only focusing on the frequency range close to the center frequency. Then the short time Fourier transformation (STFT) is applied on the filtered time signal to convert the 1D time series to 2D frequency over time signal. The hann window of $n = 4096$ samples and zero padding are applied to the segmented time windows to reduce the spectra leakage of the fast Fourier transformation. The output of the STFT contains the magnitude and phase information. Here, we only use the magnitude information to construct our Doppler profile. An example

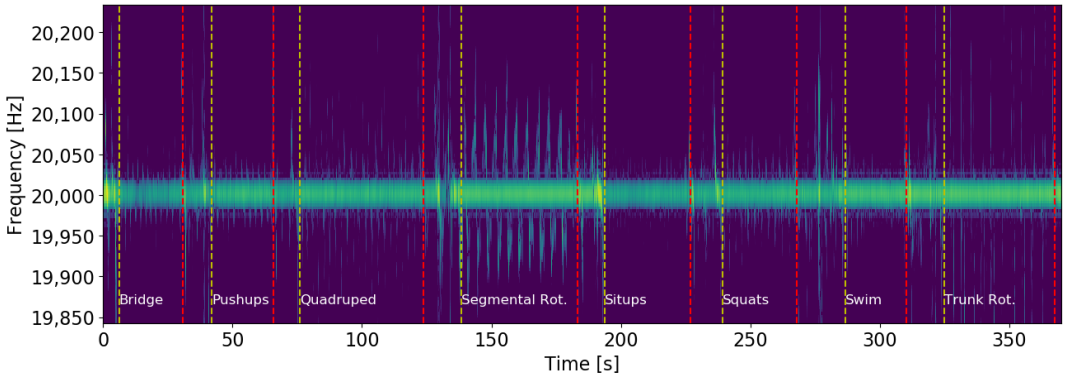


Figure 3.24.: Figure shows a sample spectrum of a full session from one arbitrary participant. The dashed lines indicate the start and the end of each workout exercise.

STFT spectrum of a full collected session from an arbitrary participant is illustrated in Figure 3.24 with the corresponding labels.

The yellow dashed line indicates the start of an exercise and the red dashed line indicates the end of this exercise. In each exercise, there are 10 repetitions included, while the swim class contains data of 25-30 s duration each. Dimension reduction is to limit the frequency bins from 19.5 kHz to 20.5 kHz. Here the maximum Doppler of 500 Hz corresponds to a maximum speed of ± 4.25 m/s. The power spectrum is normalized to the median power by applying the Equation (3.9).

$$S_{STFT} = 10 \cdot log_{10}(|X_{STFT}|^2) - 10 \cdot log_{10}(median(|X_{STFT}|^2)) \tag{3.9}$$

The segmentation part is the central part of the entire pipeline. It determines, if the segmented time window contains an exercise or not. For this task, we extracted the upper and lower envelop of the Doppler broadening in the spectrogram. Only signal with a signal variation in either positive or negative Doppler range larger than certain threshold, is supposed to contain activities. This step aims at reducing the computational load in real-time application. The time window is set to 6 s and with an overlap of 50 % for the sliding window approach. This parameter is set according to the offline processing with respect to system performance.

3.1.7.3. Classification methods

The input training samples are the segmented spectrogram with the dimension of 279×129, where 279 samples correspond to the frequency bins from 19.5 kHz to 20.5 kHz and 129 samples represent the 6 s time steps. In Figure 3.25 a sample spectrogram of each workout exercise session is depicted. These 2D spectra construct the base signal to the classifier models.

We evaluated our data on three different end-to-end neural network architectures. The first one is the 2D-CNN with randomly initialized weights and biases. The second architecture is the finetune model with VGG16 [QVF18] model as the base feature extraction layer. The last inference model is a sequence model called bidirectional LSTM. In the following section, we will introduce the model architectures and the hyperparameters used in the individual architectures. The hyperparameters are finetuned using 5-fold cross validation. The models are build using the Pytorch [PGM*19] framework and trained on a GeForce RTX 2080 module. The weights of the finetune model VGG16 is directly downloaded from the Pytorch model zoo.

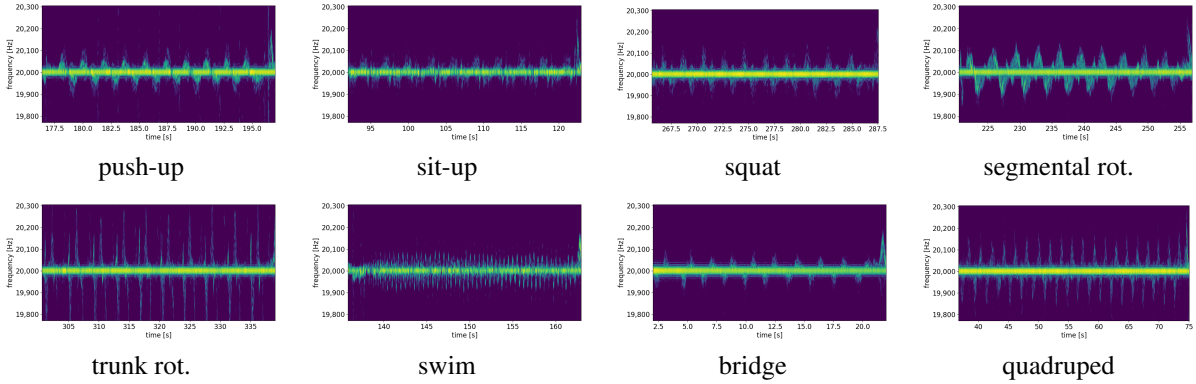


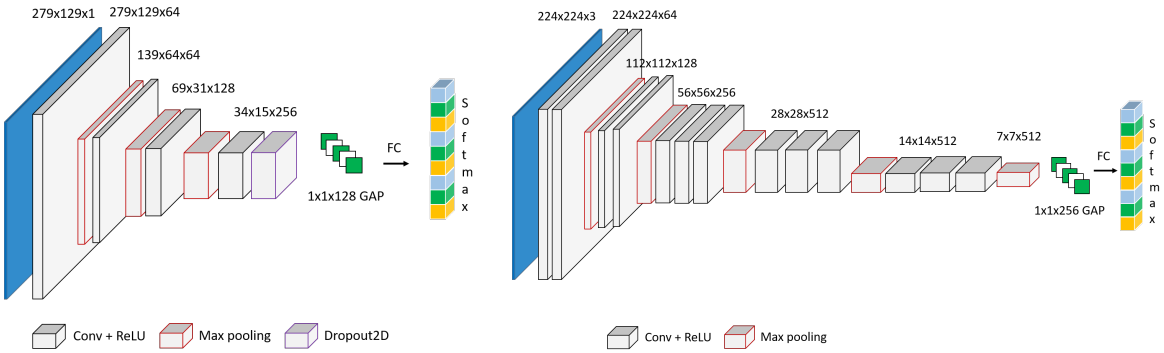
Figure 3.25.: Figure depicts the spectrogram of the sample activities. Each exercise with 10 repetitions can be easily observed. Doppler profile is distinctive around the center frequency.

2D CNN plus Global Average Pooling Layer Convolutional neural networks are best suited for automatically extracting features for classification tasks of 2D image data. Especially the pooling layers make the network architecture invariant of size variations for objects. Pooling layer can widen the field of view by sub-sampling the input image. Locally connected filters can catch local discriminant features of certain object shapes, colors and forms. The network architecture can be seen in Figure 3.26 (a). An instance normalization layer is used prior to the input in order to restrict the input range between 0 and 1 for a faster convergence.

Adam optimizer with a relatively large learning rate of 0.01 is chosen for the network to converge fast. The optimization function is to reduce the sum of the weighted cross entropy loss of the classification task. The L2 regularization for the network parameters is weighted by the factor of 0.015. The weight parameter for the classes are proportional to the class distribution. It is used in the cross entropy loss for the classification and as well as in the data sampler to constitute the training and validation batches. Each sample has its own draw probability according to the class it belongs to. A batch size of 100 is set to train for 100 epochs till the model converges.

VGG16 plus Global Average Pooling Layer We aim to improve the recognition accuracy by applying the finetune model of VGG16 network. The fine-tuning let us to exploit the base knowledge extracted from ImageNet [KSH12] task. This approach is a typical representative from the transfer learning domain by taking benefit from structural knowledge extracted from a large image pool with millions of labeled objects. The lower convolution layers are intended to automatically extract useful features for the task of object recognition in two dimensional images. We fixed the weights in the pre-trained lower feature extraction layers. The decision layer is replaced by a global average pooling (GAP) layer combined with a softmax layer to output the class probability of each exercises. The GAP layer is used to reduce the overfitting problem, due to our limited amount of input training data. The hyperparameters of a GAP layer is much smaller compared to the fully connected layer. Instead of using the $7\times7\times512$ features to the fully connected layer, we now reduced the output to $1\times1\times256$ features, which then fully connected to the class outputs with a softmax layer. The network architecture is displayed in Figure 3.26 (b).

Adam optimizer with a learning rate of 0.003 is used to minimize the cost function. The objective is to minimize the weighted cross entropy loss and the l2-regularization on the model's parameters with a weight factor of 0.015. The weight parameters for the classes are selected in relation to the class distribution as well as the data sampler. Each sample has its own draw probability according to the class it belongs to. A batch size



(a) 2D CNN + GAP

(b) VGG16 + GAP

Figure 3.26.: (a) The architecture for the 2D CNN is depicted. The final global pooling layer is used to reduce model over-fitting on the limited amount of training data. (b) It depicts the model architecture of the VGG16 finetune network with an additional global average pooling layer to reduce the model complexity and perform the classification.

of 100 is chosen and we trained 100 epochs for the model to converge. We applied an instance normalization to the input layer to restrict the input samples to the same input range. This step can be considered as another regularization step to avoid the model from overfitting. The normalization of the input data helps the model to converge faster, in which case the input range is restricted between 0 and 1.

Bidirectional LSTM Architecture The long short term memory (LSTM) model is mostly used for sequence modeling or sequence tagging [GJM13], such as natural language modeling. Recently it is adapted to work for image classification tasks as well. The network architecture of our proposed model is detailed in Figure 3.27. The architecture of bidirectional structure is rendering the network the ability to look into the future and past in order to better understand the whole context. This architecture should be able to cope better with the problem of inter-class similarity. The windowed sample spectrum is sliced to feed into the biLSTM network.

The input is instance normalized to convert the input range between 0 and 1. One important step to reduce overfitting for LSTM network is the dropout layer applied to the input before feeding to the LSTM layer. The ratio is set to 0.2 to avoid losing too much input information. This step prevents the LSTM network from simply memorizing our input data. A batch size of 100 is chosen to be trained for 100 epochs. Adam optimizer with a learning rate of 0.003 is selected to minimize the cost function. The network consists of two LSTM layers each with 128 hidden nodes. The output of the bidirectional LSTM is directly feed to a fully connected layer with the class probability as output. Gradient clipping is also applied to reduce the inherent problem of exploding or vanishing gradient from LSTM networks. The objective is to minimize the weighted cross entropy loss subjected to the underlying class distribution and the weighted l2 regularization is applied on the model parameters to avoid model overfitting. Data sampler is used to draw the batches and also corresponds to the underlying sample distribution for each class. The cross entropy loss is weighted according to the class distribution. This model architecture aims to consider the entire time sequence both in the forward and backward direction. This method compares the global structural view to the local patterns from the CNN based models.

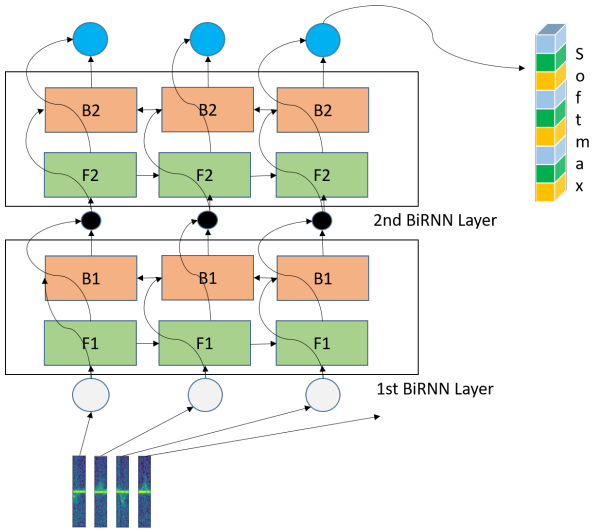


Figure 3.27.: It depicts the model architecture of the bidirectional LSTM. Each LSTM cell (B_i, F_i) contains 128 hidden nodes and two stacked layers are used to build the network. For each input node, a slice of the frequency bands (ranging from 19.5 kHz to 20.5 kHz) from a time step resolution (46.5 ms) is provided to the network.

Siamese Few Shot learning As stated before, the human activity data from sensory output is difficult to acquire in comparison to vision-based data. To overcome the problem of the small data amount, few shot classification learning is leveraged. Based on knowledge extracted on a few samples named as support samples, the network is able to generalize on similar unseen samples. This is possible under the assumption, that similar samples have similar embeddings located closer together. Here, we propose a modified Siamese network architecture to perform this task.

The Siamese network consists of two identical feature extraction base networks with shared weight parameters. We learn in general the distance between the query input against all other support samples from different classes. The class category with the closest mean distance metric towards the unknown sample is selected to be the correct class. The network architecture is illustrated in Figure 3.28. The designed structure aims to learn the optimum separation between all multiple classes at once.

The internal structure of the ConvNet is consisted of three successive convolutional layers to reduce the input dimensions. The main task of this ConvNet is to construct feature representations from input images. During the training phase, each batch consists of 9 classes and 15 query samples each class. That makes in total 135 samples per batch training. The support set is consisted of 5 support samples each class and makes in total 45 support samples. Adam optimizer is used to train the Siamese network parameter with a learning rate of 0.0005.

3.1.7.4. Evaluation

Evaluating our proposed system, we conducted a test study collecting exercise data from 14 individuals in our living laboratory. The group consists of 4 females and 10 males with an average height of 165.5 cm for the female group and 182.3 cm for the male group. Some general statistics about the test population distribution are provided in Table 4.4.

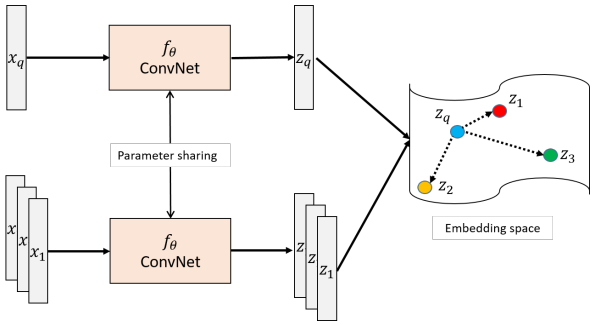


Figure 3.28.: The Siamese network is modified to adopt to few shot classification task. The structure of shared parameters for the ConvNet embeddings aims at learning the optimum separability for the multiclass classification problem by using supports. The output label corresponds to the highest similarity score towards the correct class category. Here a learned distance metric is applied to determine the similarity score.

males	age	height	**females**	age	height
Number	10	-	Number	4	-
Min	21	172	Min	21	157
Max	33	193	Max	32	172
Average	25.4	182.3	Average	24.75	165.5

Table 3.8.: It shows the statistics of the population of the test participants.

The test setup is illustrated in Figure 3.19. We placed a yoga mat on the carpet and placed our sensing device 50 cm apart from the exercising body part. The microphone of the sensing device is directly facing the exercising participant. For most of the exercises, the smartphone is aligned with the hip. Except for the *swim* and *trunk rotation*, we aligned the device with the shoulder position to better catch the micro Doppler motion from the waving arm movement. The original intention was to use a single position for all the exercises. However, the Doppler motion for exercises such as *swim* and *trunk rotation* are relatively minor at hip level due to the almost dormant hip motion. For each individual, we collected two full sessions in two successive recording sessions. Each exercise was performed ten times each in every session. Except the class *swim* was collected around 25 to 30 seconds, in order to acquire enough data samples for the sliding window approach. Otherwise, only ten repetitions are too short for the fast exercise. The participants were asked to label their data by using our recording app on the mobile device, as a way to pose less intervention on the natural action.

In the segmentation stage, we used the user-defined labels to cut periods of exercises. We further discard the first and last 1 s of each exercise at the beginning and end to remove the handling of the labeling process. The participants were asked to press a button to start and end the selected activity at the beginning and end of each exercise. The raw audio input is converted to frequency time domain by applying the STFT transformation. A sliding window of 6 s is applied to cut the spectrum for each exercise class with an overlap of 50 % for the data augmentation purpose. Since the data acquisition process for HAR is tedious and expensive, and the end-to-end training requires extensive data amount, we use overlapping segmentation to synthetically augment the input data amount. In addition, we carefully designed network regularization schemes to avoid overfitting. For each sample time window, we determine the upper and lower Doppler broadening profiles and keep only the windows with Doppler variations larger than a threshold of 2 frequency bins. This step grants us the possibility of reducing weak Doppler profiles to increase the recognition performance.

We conduct two sets of evaluation to investigate the robustness and the generalization ability of our proposed application design. Our first evaluation is conducted on the cross-subject performance. Thereby, we split the entire dataset into 70 % training and 30 % test by using a stratified splitting mechanism to maintain the same distribution of the underlying classes in both splits. For each training set, a 5-fold cross validation approach is used to fine-tune the classification models. To measure the generalization ability of our classification models on unseen test data, we keep out all sessions of randomly selected 4 individuals as holdout set to be used in the test split, while the remaining 9 individuals were used as training data. Again 5-fold cross validation is applied to fine-tune the inference models.

The evaluation metric used is the weighted F1-score. It is a better measure than precision or recall especially in the face of unbalanced class distributions. This measure provides a harmonic mean of precision and recall, compensating for the precision favors the majority and recall favors the minority class. In our evaluation, for calculating the weighted F1 score, we account the label imbalance to be included into the metric.

Cross Individual Classification As described in the previous section, a stratified 70 % : 30 % is applied on the entire dataset. The same split for the training and test dataset, as well as the 5-fold cross validation is used on the 2D-CNN, VGG16 Finetune and biLSTM models to maintain comparability across different inference models.

The weighted F1 score for the 5-fold cross validation is provided in Table 3.9. The variance across the 5-fold cross validation is quite high for 2D-CNN architecture, indicating the model's inability to learn robust and generalized features across dataset for an overall high performance. Other architectures, such as VGG16 finetune and LSTM models perform better in this regard.

Sequence modeling performs even slightly better than VGG16 finetune for our given task. The confusion matrix with the highest F1 score is shown for VGG16 in Table 3.10 and the one for biLSTM in Table 3.11. Derived from the results of the confusion matrix, we see the challenging classes, which the individual model

Method	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	σ^2
CNN	71.94 %	77.52 %	69.59 %	73.99 %	41.15 %	13.11
VGG16	81.62 %	**88.70** %	85.16 %	83.70 %	82.21 %	2.53
biLSTM	86.05 %	86.27 %	81.83 %	80.88 %	**88.86** %	2.98

Table 3.9.: For the cross subjects case, it depicts the F1 score for the 5-fold cross validation results on the three inference models.

	Predicted								
True	1	2	3	4	5	6	7	8	9
1	**.83**	.02	0.	0.	0.	.13	0.	.02	0.
2	0.	**.9**	0.	.07	0.	0.	0.	0.	.02
3	.04	0.	**.82**	0.	0.	.08	0.	.06	0.
4	0.	.06	0.	**.94**	0	0.	0.	0.	0.
5	0.	.04	0.	0.	**.96**	0.	0.	0.	0.
6	.1	.12	.1	.02	0.	**.6**	0.	.07	0.
7	0	.03	0.	.1	.03	0.1	**.73**	0.	0.
8	.08	0.	0.	0.	0.	0.	0.	**.92**	0.
9	0.	0.	0.	0.	0.	0.	0.	0.	**1.**

Table 3.10.: Confusion matrix for VGG16 in case of cross subjects training. Number corresponds to *1: bridge, 2: idle, 3: push-up, 4: quadruped, 5: segment rotation, 6: sit-up, 7: squats, 8: swim*, and *9: trunk rotation*

has difficulties to distinguish. In case of the biLSTM, the variance across the eight workout exercises is slightly smaller compared to the VGG16 model which has minor problems for interpreting similar classes. In case of the class *sit-up* in the VGG16 model, a strong misclassification tends towards the class *push-up*, and *bridge*. Both classes have a false positive rate of 10 %. This is explainable due to the similar upper body movement. Those exercises are ground-bounded while the user is lying on the ground and the sensing device is placed on the same position. Thus, the main reflections in the signal are from the same upper body part. This issue can only be resolved by considering the temporal patterns as in the biLSTM model. The sequence model biLSTM has more problems for the class *swim*. The class *swim* tends to be confused with the class *sit-up*. This is observable from Figure 3.25. The class *swim* include very small and faster arm movements. VGG16 Finetune outperforms the biLSTM model in this case about 18 percentage points on accuracy, due to its ability to observe the locally connected features. The class *squats* performs comparably worse in both models, as the distance of the performing body part is quite distant from the sensing device causing a relatively weak signal to be processed.

To be complete, we also provide the confusion matrix for the 2D-CNN model in Table 3.12. The off diagonal elements is more sever compared to the other models, indicating its inability to cope with the problem of inter-class similarity.

Generalization Ability on Holdout Individuals In this experiment, we intend to examine the generalization ability of the trained inference models. For this purpose, we select 4 individuals and use their entire sessions to composite the test dataset. The remaining 10 individuals are considered to build the training dataset. The same

	Predicted								
True	1	2	3	4	5	6	7	8	9
1	**.96**	.04	0.	0.	0.	0.	0.	0.	0.
2	0.	**.92**	0.	.02	0.	0.	0.	.04	.02
3	.09	0.	**.83**	0.	0.	.04	0.	.04	0.
4	0.	.01	0.	**.92**	0.	0.	.01	0.	.06
5	0.	.02	.01	0.	**.91**	0.	.06	0.	0.
6	0.	.04	.09	0.	.02	**.76**	.04	.04	0.
7	0.	.06	0.	0.	0.	.22	**.72**	0.	0.
8	0.	.03	.06	0.	0.	.16	0.	**.74**	0.
9	0.	0.	0.	0.	0.	0.	0.	0.	**1.**

Table 3.11.: Confusion matrix for biLSTM in case of cross subjects training. Number corresponds to *1: bridge, 2: idle, 3: push-up, 4: quadruped, 5: segment rotation, 6: sit-up, 7: squats, 8: swim*, and *9: trunk rotation*

	Predicted								
True	1	2	3	4	5	6	7	8	9
1	1.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	.8	0.	.1	.05	0.	0.	.04	.01
3	.24	0.	.49	0.	0.	.17	0.	.1	0.
4	0.	.04	0.	.84	0.	.09	0.	.03	0.
5	0.	0.	0.	0.	1.	0.	0.	0.	0.
6	.02	0.	0.	.13	0.	.57	0.	.28	0.
7	0.	0.	0.	.22	.04	.26	.35	.13	0.
8	.17	0.	.03	.02	0.	.17	0.	.59	.02
9	0.	.01	0.	.01	0.	0.	0.	0.	.97

Table 3.12.: Confusion matrix for 2D-CNN in case of cross subjects training. Number corresponds to *1: bridge, 2: idle, 3: push-up, 4: quadruped, 5: segment rotation, 6: sit-up, 7: squats, 8: swim*, and *9: trunk rotation*

Method	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	σ^2
CNN	68.93 %	59.38 %	67.85 %	60.73 %	72.66 %	5.06
VGG16	**77.96** %	75.98 %	74.69 %	71.31 %	74.52 %	2.17
biLSTM	75.23 %	76.96 %	**81.37** %	76.91 %	79.52 %	2.17

Table 3.13.: For the generalization test with holdout test participants, it depicts the F1 score for the 5-fold cross validation results on the three inference models.

	Predicted								
True	1	2	3	4	5	6	7	8	9
1	**.62**	0.	.09	0.	0.	.19	0.	.1	0.
2	.02	**.67**	0.	.04	.01	.15	.02	.01	.08
3	.09	0.	**.54**	.07	0.	.25	0.	.05	0.
4	0.	.04	0.	**.68**	0.	.09	.15	0.	.04
5	.03	.01	.01	0.	**.93**	0.	.01	0.	0.
6	.02	0.	.12	.02	0.	**.76**	0.	.07	0.
7	0.	0.	0.	0.	0.	.22	**.67**	.11	0.
8	0.	.03	0.	0.	0.	.1	0.	**.87**	0.
9	0.	.02	0.	0.	0.	0.	0.	0.	**.98**

Table 3.14.: Confusion matrix for VGG16 in case of holdout subjects training. Number corresponds to *1: bridge, 2: idle, 3: push-up, 4: quadruped, 5: segment rotation, 6: sit-up, 7: squats, 8: swim*, and *9: trunk rotation*

5-fold cross validation split is applied across all three inference models to maintain comparability of the model performance. The class distribution in the training and test dataset are closely equal.

The evaluation result is provided in Table 3.13. The expected performance drop is observed in this specific setup. This performance drop is explainable by the diversity of the collected data. Since human activity, especially the targeted sport exercises, is highly complex and diverse, we need lots of diverse data to train a model, which can cope with all possible situations. This is hardly possible.

The corresponding confusion matrix for the VGG16 and biLSTM network are depicted in Table 3.14 and Table 3.15. The biLSTM model caught the structural sequence information better than the VGG16, since the off-diagonal elements are slightly smaller than in case of the VGG16 finetune model with the exception of certain classes, such as *sit-up* and *swim*. Several exercise classes are quite similar, such as *sit-up* and *bridge* as can be seen in Figure 3.25. The VGG16 model with the pooling layer thus makes it harder for the network to distinguish both classes, while the sequence model without modifying the sequence information is still able to distinguish both exercises. However, the overall performance for both models is quite similar compared to each other. The trend for the performance on each class reflects the results from the cross-subject method discussed in the previous subsection.

In Table 3.16, the confusion matrix for the 2D-CNN for the holdout test dataset is shown. The model is unable to generalize for several classes as the diagonal elements of the confusion matrix are quite low for certain exercise classes. The class *sit-up* and *squats* only perform slightly better than uniform distribution. This is also the case for the cross-subject approach.

	Predicted								
True	1	2	3	4	5	6	7	8	9
1	**.79**	.01	.13	0.	0.	0.	0.	.07	0.
2	0.	**.81**	0.	.12	0.	.03	0.	.04	0.
3	.09	0.	**.7**	0.	.04	0.	0.	.17	0.
4	0.	.13	0.	**.82**	0.	0.	.04	.01	0.
5	.04	0.	.01	.03	**.91**	.01	0.	0.	0.
6	0.	.23	.07	0.	0.	**.46**	.05	.19	0.
7	0.	.05	0.	0.	.09	.14	**.73**	0.	0.
8	0.	.03	0.	0.	.03	.21	0.	**.73**	0.
9	0.	0.	0.	.01	0.	0.	0.	0.	**.99**

Table 3.15.: Confusion matrix for biLSTM in case of holdout subjects training. Number corresponds to *1: bridge, 2: idle, 3: push-up, 4: quadruped, 5: segment rotation, 6: sit-up, 7: squats, 8: swim*, and *9: trunk rotation*

	Predicted								
True	1	2	3	4	5	6	7	8	9
1	.87	0.	.08	0.	0.	.03	0.	.03	0.
2	.07	.47	0.	.26	.02	.03	0.	.01	.13
3	.42	0.	.54	0.	0.	0.	0.	.04	0.
4	.01	0.	0.	.86	0.	.1	0.	0.	.02
5	0.	.05	.05	0.	.9	0.	0.	0.	0.
6	.4	0.	.12	.1	.02	.19	0.	.17	0.
7	.2	0.	.07	.2	0.	.4	.13	0.	0.
8	.25	.04	.18	.14	0.	0.	0.	.39	0.
9	0.	0.	0.	.06	0.	0.	0.	0.	.94

Table 3.16.: Confusion matrix for 2D-CNN in case of the holdout subjects training. Number corresponds to *1: bridge, 2: idle, 3: push-up, 4: quadruped, 5: segment rotation, 6: sit-up, 7: squats, 8: swim*, and *9: trunk rotation*

Method	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	σ^2
Siamese cross	93.50 %	94.11 %	93.77 %	93.62 %	93.87 %	0.21
Siamese holdout	85.70 %	86.04 %	85.41 %	85.50 %	85.16 %	0.30

Table 3.17.: Classification accuracy for both setups are listed here. For the evaluation, 5 support samples each class are used. Compared to other proposed classifiers, we observed an increase of at least 7-10 percentage points. The reason is because the few shot classification task by including knowledge from a few known samples is especially suited for learning with limited data amount.

Error Measure	bridge	push-up	quadruped	seg rot	sit-up	squat	trunk rot
mean overall	1.04	1.29	1.18	2.64	2.86	1.79	1.45
standard deviation	1.15	1.41	0.89	1.63	1.41	2.01	1.18

Table 3.18.: Mean error count compared to the reported ground truth count is given for the 14 test participants.

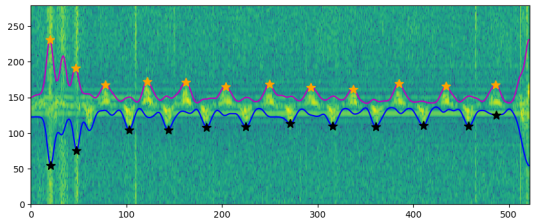
Results on Few-shot learning We noticed that we can further increase the classification performance by leveraging few shot classification learning method. In this work, we examined the model generalization ability by including few unseen samples. With only 5 support samples from each class during classification, we can improve the final performance on unseen test data. In both individual experimental setups (cross-subject case and with holdout users), we observe an increase of more than by 7-10 percentage points in average for both best working models proposed in this work (VGG16 and biLSTM) and even 20 percentage points for the worst model architecture (2D CNN). The results are listed in Table 3.17.

3.1.7.5. Counting

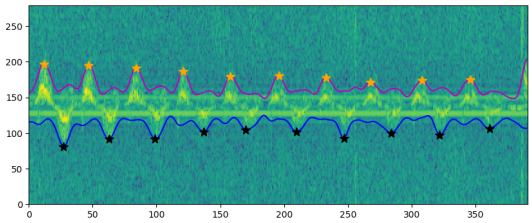
Exercise counting can be viewed as a parallel task after classification. Here a short conceptual view of the counting method is proposed. For counting the exercises, we extract envelopes from both the positive and negative Doppler profiles from the pre-processed spectrogram. The envelop stretches from the middle frequency component and broadens to both directions as the amplitude falls below a minimum threshold. The envelop signal is further smoothed with a Gaussian kernel of size 3 to suppress noisy signals. A simple peak detection algorithm is applied on the Doppler envelop with finetuned minimum peak distance for suppressing multiple peak detection and to ensure clear separable peaks from the exercises.

For exercises with left and right variations, such as *quadruped* and *trunk rotation*, the negative envelop of the Doppler profile performs better compared to positive envelop. Due to micro-Doppler motion from the arms and legs, multiple peaks is detected for one repetition. As for *quadruped, segment rotation* and *trunk rotation*, a high peak followed by a lower peak is resolved as one repetition, since the high peak indicates the main reflection, while the lower peak represents the remote Doppler movement from the opposite body part. As for clear defined repetitions such as for *bridge, push-up, sit-up* and *squat*, both the positive and negative envelops can be used to count repetitions. In Figure 3.29, a conceptual view of the exercise counting can be viewed with the asterisk indicating the detected peaks of the repetition.

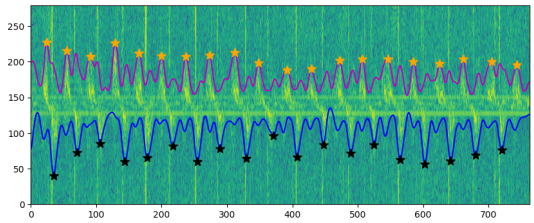
In Table 3.18, the mean error count and the standard deviation compared to the reported ground truth count for the given 14 test participants are provided for each exercise. In Figure 3.30, the mean counting error in relation to the reported true count are depicted for each test participant individually (marked with a cross) and the mean error across all test participant is marked with a diamond symbol.



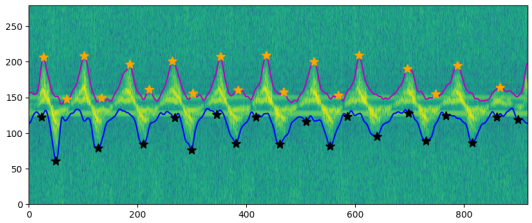
(a) Bridge



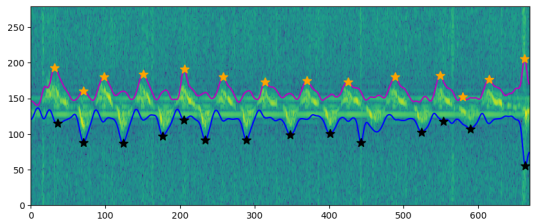
(b) Push-up



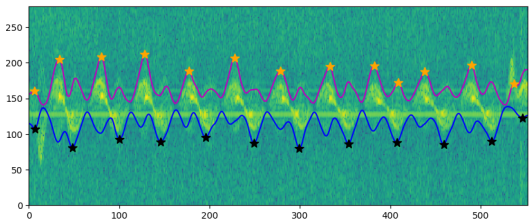
(c) Quadruped



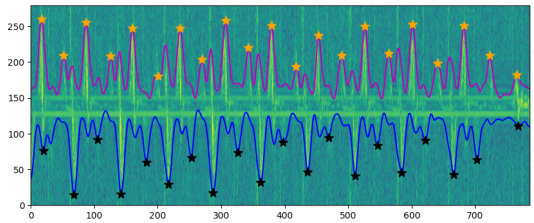
(d) Segment Rotation



(e) Sit-up



(f) Squat



(g) Trunk Rotation

Figure 3.29.: For each exercise, the positive and negative Doppler profiles are depicted. A simple peak detection algorithm is applied to detect the number of repetitions. There are 10 reported repetitions in each exercise. For exercises with left and right variation such as *quadruped, segment rotation* and *trunk rotation*, the negative Doppler profile performs better with higher maxima followed by lower maxima caused by micro-movement from the limbs and arms.

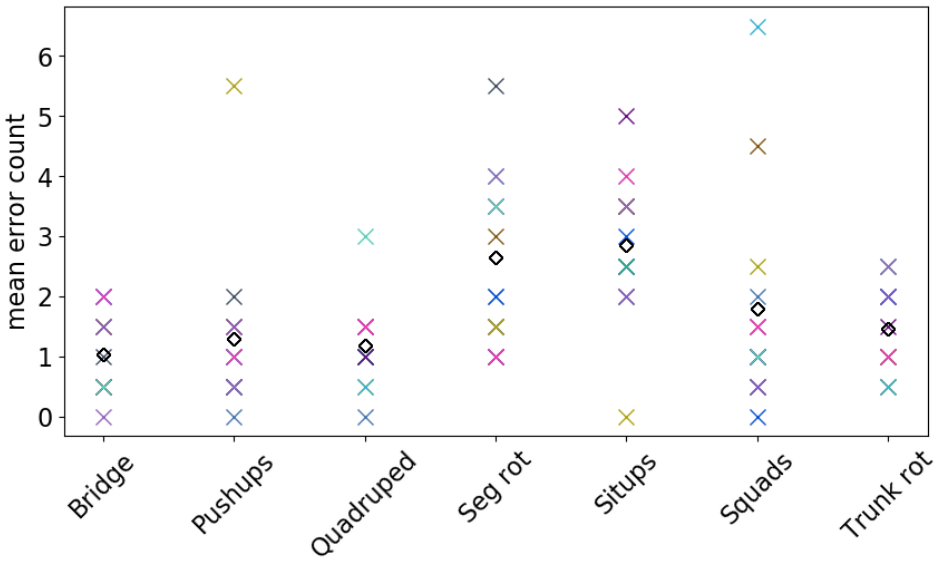


Figure 3.30.: For each exercise, the mean counting error for each test participant is depicted with a cross, where each color represents the ID of the individual test participant and the mean error count over all test participants is depicted with a diamond symbol.

3.1.7.6. Discussion

Since Quantified-self can lead to a healthy life style, we propose an novel application of using a unmodified commercial smartphone to recognize eight whole-body workout exercises. The set of activities are chosen to reflect a more realistic setting. Our application aims at mobile and remote sensing to enable practice everywhere at anytime. By using the integrated hardware of the smartphone, we turn it into an active sonar device to measure the Doppler profile caused by moving body in the vicinity of the sensing device. Carefully designed processing and segmentation steps help us to work with even weak reflections. However, we identified several challenges during our evaluation phase, which can be improved in future applications.

First, we encountered the physical limitation of occlusion. The placement for certain exercises is not ideal for receiving the micro Doppler information originated from the legs or arm motion. The current distance of 50 cm apart from the hip level is too far, so the back reflection is sometimes relatively weak due to a strong attenuation. This proposed application is thus best suitable for short range application within a distance below 50 cm. This distance measure is according to the performance test conducted in Section 3.1.7.1 regarding the signal strength in relation to distance.

Second, the Doppler profile for several classes are very similar. This is the problem of inter-class similarity. To cope with this problem, we identify the stacked bidirectional LSTM model to be more appropriate. The shape and rotation invariance introduced by the pooling layer of convolutional model makes it sometimes even more difficult to distinguish between these classes, as can be seen in the higher off-diagonal elements from the confusion matrix in the evaluation section 3.1.7.4.

Finally, the inter-person variability caused a relatively strong performance drop as observed in the holdout subjects for testing case. This problem is inherent to the complex nature of human activities. Different people

show different affinity towards physical exercises. People regularly perform workouts intuitively have a different signal shape than those who do not perform sport on a regular basis. Careful design should be applied to resolve this challenging issue. Ensemble models can be leveraged for different groups of users. We could first cluster different users into similar groups with its individual classification model. Then subject to the outcomes from the ensemble learning, the final decision can be fused using an appropriate weighting scheme.

Another way to address the problem of inter-person variability or complexity in data is to use few-shot classification learning task. By leveraging the modified Siamese few-shot learning classification, we improved the overall performance in both experimental setups by at least 8-9 percentage points in average for both best working models. Especially in the holdout experiment, by only including a small portion of the unknown samples, we achieved a large increase by 7-10 percentage points in the classification performance. Usually, we can not train conventional deep learning methods by applying this few samples. In this case, we benefit from the objective of the few-shot classification by increasing the generalization ability through knowledge extracted from support samples from different categories. By mapping objects from the same category closer together in the embedding space and measuring a metric distance, unknown object from the same category can be easily determined in comparison to other categories.

3.1.7.7. General findings

In this Section 3.1.7, we showed the first results of using a commercial smartphone (in this work, we used Samsung Galaxy A6 2018) to remotely detect eight more realistic and complex whole-body exercises. We leverage the Doppler motion profiles, caused by human motion and especially the micro Doppler motions caused by the limb movement to catch the delicate features across the eight exercises. The aim is to build a mobile application allowing the user to practice anywhere at anytime without the need to carrying any extra hardware setups or wearables.

We performed a thorough investigation on the carefully designed inference models, by using a 5-fold cross validation on cross-subject dataset and the holdout dataset. In case of the holdout set, we intend to examine the generalization ability of the used inference models on data from new participants. This case resembles the use-case in reality, where the ability to adapt to new users is important. Sequence modeling structures, such as the bidirectional LSTM network shows good performance in both evaluation setups. Closely followed by VGG16 finetune network, we see its ability to transfer its power in low level feature extraction to user specific tasks. According to the biLSTM network, we achieve the highest F1 score of 88.86 % for the cross subject case and 79.52 % for the holdout participants. VGG16 presents similar good results with 88.70 % for the cross subjects case and 77.96 % for the holdout dataset.

Few shot classifier further improves the performance by more than 7-10 percentage points in average for both best working models and even 20 percentage points for the worst model architecture, in both experimental setups by leveraging the metric information in the feature embedding space. Only with the knowledge extracted from few support samples, the generalization on samples from similar classes is possible to achieve.

The design choice of choosing convolutional neural network (CNN) as the feature extraction stage preferred to conventional classifiers working on features engineering is clarified in the following. Conventional classifiers, such as support vector machine (SVM), or decision trees (DT), build upon the handcrafted features that still need to be extracted from the spectrum domain. This presents additional computational and design efforts. In addition, the performance of the model is strongly dependent on the handcrafted features, which can be quite constrained due to the inductive priors introduced. CNN can work directly with 2D spectrogram, without further expert knowledge to design the appropriate and discriminative feature representations. The class separability is integrated into the optimization process itself. Therefore, we assume this approach to work well on our 2D

Doppler profiles. As the current model can be trained on a more powerful server, the smartphone only needs to host the model for inference, which should be able to work fast.

In the previous section, we focused on using off-the-shelf commercial hardware to design applications for recognizing human activities. Commercial hardware enables fast prototyping, as it reduces the overhead of hardware design and can be fully integrated into existing infrastructure. But at the same time, commercial hardware also means restrictions on the degrees of freedom to freely set system design parameters. In the next section therefore, we will lift the restrictions on hardware limitations by freely designing our own hardware prototype to recognize the same set of human actions to enable a better cross comparison of both approaches addressing the research questions related to building a successful HAR application. In the next section, we introduce a customized prototype by applying capacitive proximity sensors for ubiquitous dynamic human activity recognition.

3.2. Active electric field sensing

Capacitive technology can be found in almost every touchscreen technologies for common smartphones or tablet PCs today. Besides touch modality, capacitive sensing can be extended to measure various other physical properties, such as proximity, acceleration or deformation. Capacitive measurement is easy to apply, light weight and power efficient. These positive features make it a promising research topic to build smart appliances for human machine interface, ranging from posture recognition, indoor positioning system to behavioral analysis.

The history of capacitive sensing dates back in the early 19th century. Léon Theremin invented a musical instrument in 1917 called Theremin [Gli00], which is the first electronic musical instrument played without direct physical contact by the performer. In the last two decades, capacitive sensing plays an increasingly important role in the research field of human computer interaction. It touches various application domains.

Smart Objects: The form factors of a capacitive sensor can be small and can be integrated into everyday objects. This renders the enhanced object the opportunity of being "smart" and able to interact with people. Sato et al. [SPH12] proposed to use a novel swept frequency capacitive sensing technique to not only detect touch interaction, but also enable more complex configurations of interaction. Integrating this technique into a door knob, they were able to distinguish between *no touch, one finger touch, pinch, circle*, and *grasp*. Embedding this technique under a wooden table, they were capable of recognizing various postures, such as *not present, present, one hand on the table, tow hands, one elbow, two elbows*, and *arms placed flat* on the table.

Indoor Localization: Capacitive sensors can sense the presence of any grounded objects and humans. Embedded under any non conductive flooring material, it can be used as a tag-free localization system. Steinhage et al. [STS*13] built a smart floor system by using modular loading capacitive sensing units under the floor carpets. The modular units enable easy exclusion of areas, where the sensing area is occluded by furniture, but needs more sensor units to achieve a high sensing resolution. The capacitive floor can detect the presence of human and track them by keeping their trajectories. Possible assisted applications such as gait analysis, recognition of activities of daily living, behavioral tracking and fall detection are implemented among others. Opposed to the modular setup, Braun et al. [BHW11] propose a smart floor by using a grid like layout. For this installation, the sensors are placed on the edge, while the sensing electrodes are laid under the floor covering with a spacing of 20 cm. Electrode wires are more robust and a malfunctioning sensor can be exchanged without removing the floor covering.

Other then placing the sensor on the ground, they can also be installed on the wall to enable interactive play or to control smart home appliances. Zhang et al. [ZYH*18] developed an interactive wall by leveraging the mutual capacitance sensing. They covered an entire wall in a grid like layout. The mutual capacitance enables them to measure on each intersection point and provide thus a fine resolution of the room scaled sensing area.

Compared to loading mode sensors, mutual capacitance enables $N \times M$ measurements by only applying $N + M$ electrodes using time multiplexing, while for the loading mode sensor $N \times M$ sensors are required to have the same resolution. The wall can sense gestures and even perform pose estimation, if the human body is standing closely in front of the wall. Making use of the electric noise of electric appliances, the wall can even recognize multiple electric appliances in use. The system enhances the original functionality of a wall and makes it to offer a rich source of context sensitive information.

Posture Detection: Posture detection is one of the most commonly applied domains by using capacitive sensors. Since the capacitive measurement senses the change in a quasi-static electric field, it is most suitable for providing stationary information. Valtonen [VMV09] used the human body as active antenna to make the underlying system to track whole-body poses. The sensing system consisted of multiple transmit electrodes of different sizes embedded under floor tiles and a receiving electrode placed on the wall. The human body can be used to couple the low frequency signal generated by the underlying transmit electrodes to the receiver. This induced current on the receiver side can be measured to infer the body distance and body pose.

The sensing electrode can be build of different materials, such as solid copper plates, shielded or not shielded electrode wires, or even flexible conductive textiles or garments. The variety of the sensing electrodes encourages researcher in building diverse applications with this sensing technology. Braun et al. [BFMW15] integrated thin copper plate electrodes into the cover of a driver's seat to measure the sitting postures or the level of fatigue of the driver. In accordance to the periodical modulation in the capacitance value caused by the chest movement, they are able to measure physiological signals, such as the respiratory rate. In the project *Jacquard*, Poupyrev et al. [PGF*16] presents the design and manufacturing technologies of smart textiles built with conductive yarn or garments. The smart textile with touch sensitive areas can be used to activate different services or provide functionalities, such as to start Google Maps on your personal phone, pick up a call or turning on and off a music app.

Opposed to the common stationary use-cases of the active capacitive sensors, such as indoor localization which aims mainly to detect the presence and absence of the inhabitants, or human posture recognition tasks, I would like to explore its ability on more dynamic activities, such as the whole-body exercises such as repetitive workouts for Quantified-self applications.

3.2.1. Introduction

For the last decades, research on capacitive sensing makes more advances in ubiquitous sensing, wearable and stationary devices. A great benefit of capacitive sensing is, that it can offer highly interactive system designs at low cost. In contrast to vision-based systems, capacitive measurement is purely time series. Vision provides more contextual information but could impair privacy in private or public domains. Therefore time series can help us to build sparse implementation, while maintaining privacy. The design choice of the electrode materials or shape determines the detection range, sensitivity and resolution.

Rapid prototyping tool makes building capacitive prototypes easier compared to radar-based applications where customized hardware is required. OpenCapSense is an open-source rapid prototyping tool for capacitive applications designed by Grosse-Puppendahl [GPBB*13]. It enables us to develop new types of pervasive user interfaces with minimal effort. It can be configured to sensor update rates of up to 1 kHz and has a spatial resolution of 1 mm at close distances, and around 1 cm at distances greater than 35cm. The board consists of a circuit board with a microcontroller and interfaces for up to eight sensors that can be connected via serial bus connections. The data transfer to a processing unit can be done over the CAN or I^2C Bus. This board can be operated in all three capacitive modes, including loading mode, shunt mode and transmit mode sensors. Wimmer [WKBS07] also proposed another open-source capacitive sensing toolkit for pervasive activity recognition

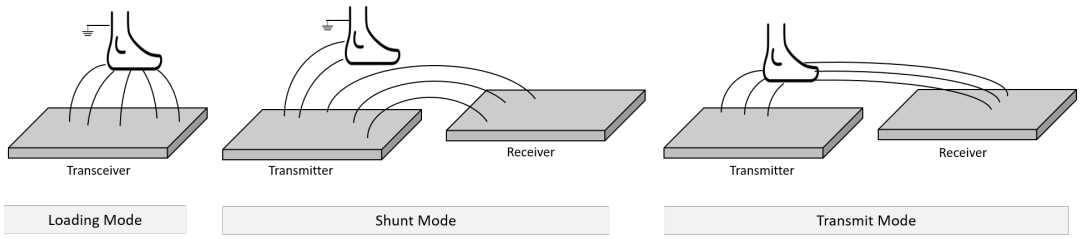


Figure 3.31.: Figure depicts the three operation modes of an active capacitive measurement [Smi96]. In loading mode, the electrode acts as both transmitter and receiver. Any approaching conductive object results in an increased capacitive coupling. In shunt mode, the electric field strength decreases, if an object is inside the sensing area due to partial occlusion of the electric field lines. In transmit mode, the electric field strength increases, if an object is present. In this case, the object acts as an extended transmitter and shortens the distance toward the receiver electrode.

and detection. The availability of these open-source toolkit encourages researchers to quickly build prototypes, applications and algorithms to explore their ideas.

Large-scale capacitive systems, such as floor-based indoor localization systems, need large power supply to operate. Thus these installations are not mobile and require large processing and storage units to access the data. Wearable capacitive devices have the restriction of the user to wear the device directly on the body, in order to achieve good performance. Flexible conductive textiles suffer from the problem of deformation. The deformation may cause the received signal to change its shape and leads to drop in performance of data-driven models. Therefore, my research interest in capacitive sensing is on developing portable, small-scaled systems, which have the ability to perform remote sensing by leveraging the proximity sensing ability of active capacitive sensors. Despite the portability, the system should remain form stable and robust towards environmental changes.

Regardless of the variety in sensing modalities of capacitive measurement, we focus on the investigation of loading mode sensors to build sparse and real-time capable applications. In the study in Section 3.2.3, we integrated proximity sensing ability to a commercial yoga mat to recognize eight workout exercises. With the sparse setup of using only eight capacitive measurements and inference model with moderate capacity, the application can be deployed on a Raspberry Pi 3 [RW12] device in real-time.

3.2.2. Physical Principles of Capacitive Sensing

Typical active capacitive measurement can be operated in three different modes, such as loading mode, transmit mode and shunt mode according to the definition by Smith [Smi96]. The working principle is illustrated in Figure 3.31. The quasi-static electric field modulation is measured by the capacitive coupling between the transmitter and receiver electrode. A low frequent electric field is generated between the transmitter and receiver node. The quasi-static field strength is affected by the presence of a non-conductive object between the transmitter and receiver. The loading mode operation form is the most simple case of all three operation modes. In this case, the transmitter and receiver are the same sensing electrode. If no grounded object is present, the field lines are weak, since you can image an object placed in an infinite distance. If a grounded object is present, the capacitance increases with the decreasing distance coupled with the term of $\frac{1}{d}$.

In shunt mode, electric field lines go from transmitter to the receiver. When a grounded object is present between the transmitter and receiver, the electrical field lines are partially occluded causing the original electric

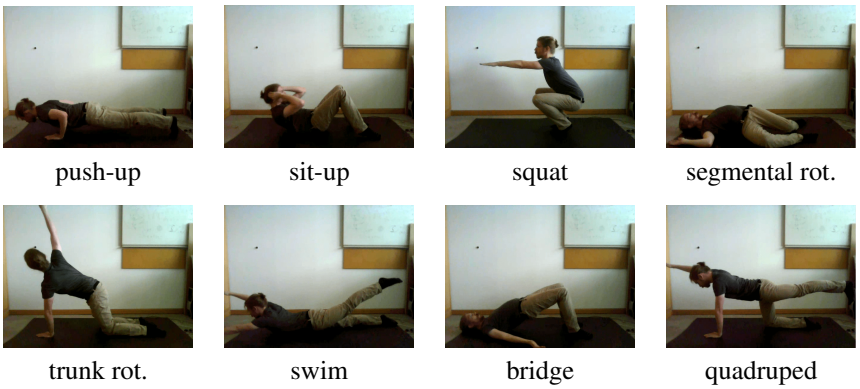


Figure 3.32.: We collected the eight exercises in an office environment. The enhanced yoga mat is placed in the middle of the room. Figure depicts the eight exercise classes.

field strength to decrease. In this operation mode, we can achieve maximum measurement points with minimal sensing electrodes by leveraging the time multiplexing approach. Each sensor node can be operated both as a transmitter or receiver. Thus having a grid of $N + M$ sensing electrodes, we can have $N \times M$ measurements. This operation mode can be used to have fine sensor resolution with minimum sensor hardware. Though the effort lays in the time scheduling approach to set up the correct measurement.

The transmit mode is similar to the shunt mode, except instead of occlude the electric field lines, the human body acts as a transmitter by coupling the electric field lines from the transmitter and conduct it further to the receiver side. When the body is distant to the receiver, the electric field strength is weakened with the term $\frac{1}{r^2}$. When the body is close to the receiver, the electric field strength weakens with the term $\frac{1}{r}$.

Since the loading mode is the most simple operation mode and it is most suitable to measure distance profile from a distant grounded object, our following prototypical application is based on this sensing technique for having minimum design effort to achieve real-time applications.

3.2.3. Study: Whole-body exercise recognition with proximity capacitive sensing

Applications for Quantified-self task are mainly targeted in this chapter. In this section, I present an alternative sensing technology to detect the same set of exercises used for ultrasonic sensing with a commercial smartphone. In comparison to the previous sensing technology, the capacitive sensing shows certain assets. Since the proposed system is a self-designed prototype, the parameters setting is configurable and independent of fixed hardware device. They can be adapted to the task. The capacitive proximity sensing can also overcome the common issues of other sensor technologies commonly used for this task, such as wearable or pressure-based sensing. Details will follow in the discussion section. In this section, we base our results and findings from our work published in [FJKK20] to recognize eight fitness exercises: *push-up, sit-up, squat, segmental rotation, trunk rotation, swim, bridge* and *quadruped*. A list of the targeted exercises is illustrated in Figure 3.32. The prototype is built with capacitive loading mode sensors.

The set of exercises are chosen such that we can show the advantage of our floor-based application compared to wearable devices. Further justification of choosing this set is to focus on whole-body exercises that can demonstrate the power of proximity sensing over pressure sensing. In case of *push-up* class, the chest movement

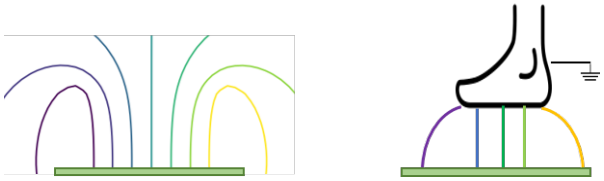


Figure 3.33.: Figure approximates the electric field lines in the case of loading mode capacitive sensing. If no object is present, the opposite parallel plate can be imagined to be in the infinity. When a human object is above the plate capacitor, the electric field lines will end on the foot as the opposite plate capacitor. Electric field lines are always orthogonal to the surface.

to and away from the sensing surface are still visible in the signal without touching. Thus the contribution of our work are as follows:

- We propose a floor-based sensing system using capacitive proximity sensors to recognize eight strength-based sport activities.
- Due to the limited amount of labelled training samples, we demonstrated the ability of using data augmentation methods to regularize the classification network from overfitting.
- We further demonstrate the advantages of using our proposed floor-based system to a single arm worn accelerometer or a single ultrasonic mobile sensor for the targeted set of whole-body activities.
- Due to the simple network model architecture and the sparse resolution of our system, it can be run online on a Raspberry Pi 3.

3.2.3.1. Hardware prototype

The used yoga mat is a common consumer mat. Its dimensions are $190 \times 100 \times 1.5$ cm and it is made of synthetic, soft and form preserving rubber. The OpenCapSense (OCS) toolkit [GP] is used to develop the *ExerTrack* prototype. The capacitive operation mode for our proposed application is called loading mode. In this operation mode, one side of the sensing electrode generates a low frequent electric field. The distance of the single sided electrode plate and the human body is measured through the change in capacitance coupling over the electrode to the ground. The operation principle is illustrated in Figure 3.33.

If no object is present in the electric field, the other plate is thought to be in a distance of infinity. If an conductive object is present within the sensing area, the capacitive coupling between the conductive object and the sensing electrode will be measured. This affects the loading characteristics of the capacitor. The coupling capacitance is measured by averaging the measurement for the charging and discharging time of the capacitor over a small time window. Averaging sensor values over time reduces the noise coupled to the input signal. To further reduce the environmental coupling from the ground surface, the capacitive sensor applied in this work is able to use an active shield on the ground side. The function of the active shield is to load both the sensing plane and the opposite shield plane with the same electric potential. In this way, there is no direct coupling from the sensor electrode to the ground. This should reduce the parasitic coupling from the ground to the sensor. Thus by applying the active shield, the different ground surface should not affect the sensor measurement. This positive aspect of active shielding is depicted in Figure 3.34.

The operation frequency of our proposed sensing mat is at 20 Hz. According to literature research, we identified, that the human body movements are contained within frequency components below 20 Hz and 98 % of

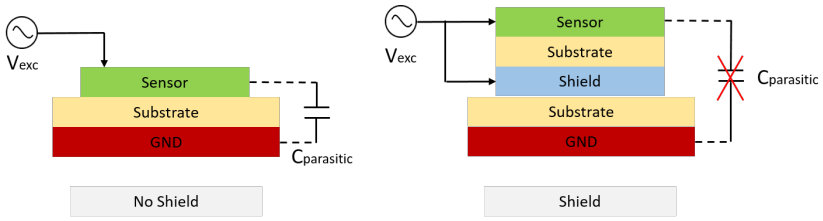


Figure 3.34.: It depicts the working principle of an active shield to avoid the parasitic coupling from the environment. In case no shield is used as depicted in the left panel, there exists a parasitic capacitance acting as an additional noise source. To mitigate this effect, an active shield is applied to load the sensor and shield to the same electric potential.

Property	double sided plate	sheet	foil	cable
Shielded	yes	no	no	no
Range	15 cm	5 cm	3 cm	0 cm
Robustness	rigid	soft	deform-able	solid
Signal quality	very good	noisy	noisy	good
Scalability	very good	good	bad	good
Longevity	robust	short	short	robust

Table 3.19.: Comparison of electrode materials. The option of shield is of vital importance to increase the sensing range and robustness. Therefore the choice of use in this work is the double sided copper plate with sensing and shielding side.

the information are contained within the Fast Fourier Transformation (FFT) coefficients below 10 Hz [Sho15]. Thus this operation frequency of 20 Hz is sufficient to operate for the intended tasks of whole-body sport activity recognition and keeps the processing overhead low.

Different electrode materials, including copper wire, conductive textiles, copper stripes and copper plate were tested with respect to their sensitivity, signal strength and dynamic range. We decided to use shielded solid copper plates, since it permits robust detection of floating body parts up to 15 cm above the mat. Details are provided in the Table 3.19. With regard to the sensing range, robustness and signal quality, we observed the importance of active shield. It aims at mitigating parasitic and environmental inferences and thus increases the signal to noise ratio for the true measurement. Therefore the choice of using double sided copper plate is justified.

The size of the double sided copper plate as electrode was chosen with respect to the coverage of the sensing area and the detection range of distant object. The larger the electrode surface, the larger is the detection range with the cost of more power consumption to load the plate electrode. Testing different electrode positions, Velcro tape was sewed on a blanket and glued on the shielded side of the plate electrodes. This enables testing different sensor placements while ensuring that the electrodes do not change position after continued use of the mat. As such, the mat itself remains portable and the blanket with the electrodes can be attached to other yoga mats. This is mainly designed for reliably testing the prototype and is not intended as a commercial prototype. The prototype of our design can be seen in Figure 3.35 left.

Regarding the sensor placement, initially the idea was to cover different body parts with the sensors. However in order to reduce the constraints on the user's orientation and posture, we decided to use the symmetrical layout depicted in Figure 3.35 right. This symmetrical layout can be easily leveraged for creating synthetic data, also

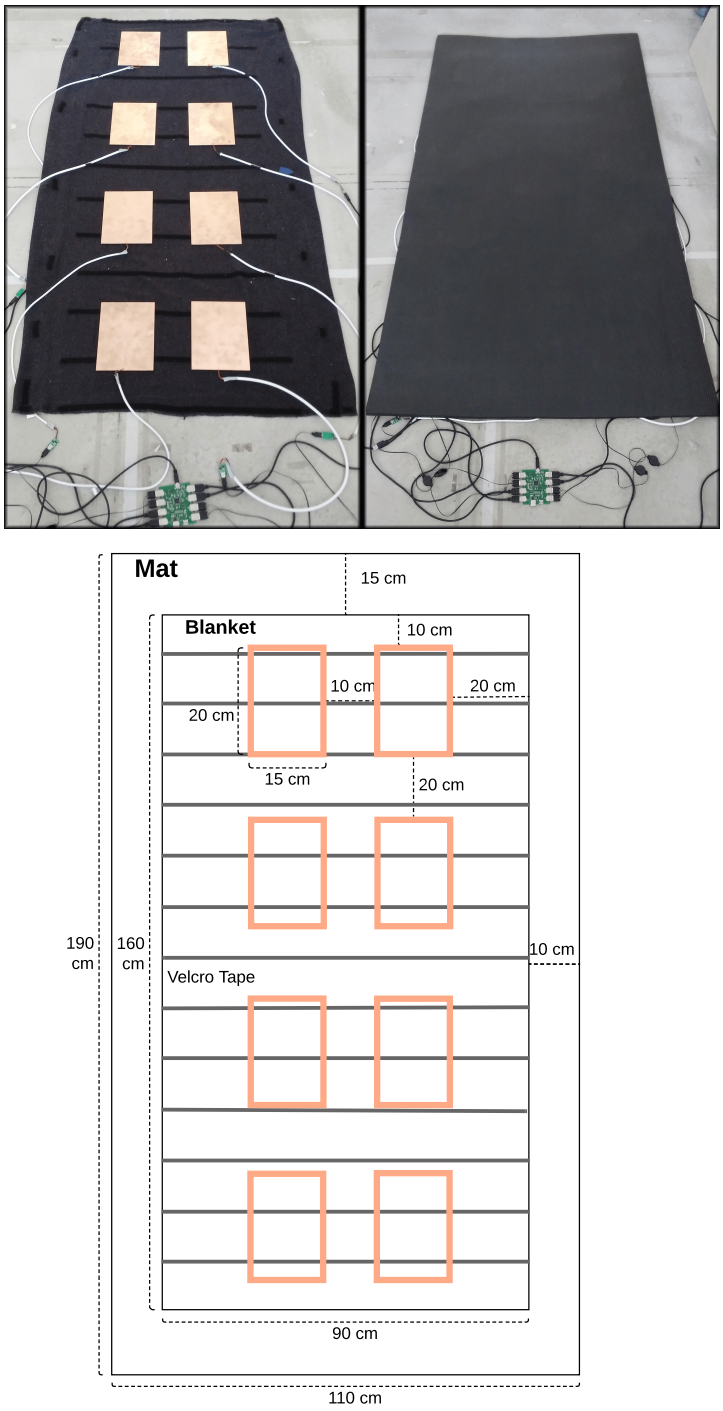


Figure 3.35.: Figure illustrates the prototype of our proposed sensing mat for sport exercise recognition by using eight capacitive proximity sensors. The symmetrical setup is used to enable the option of easy data augmentation process.

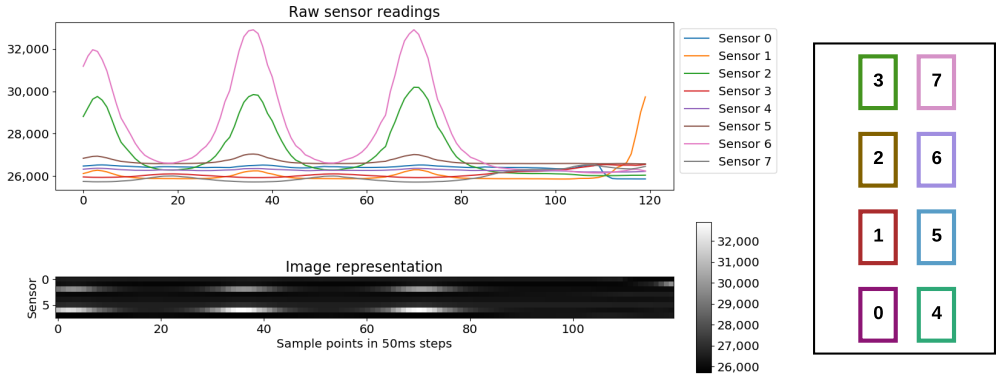


Figure 3.36.: A raw sensory input of 6 seconds length and its corresponding sensor values is depicted. The image representation is just another representation of the same information as time series. The sensor numbering is depicted in the right panel.

known as data augmentation. This setup enables us to simply rotate and mirror the recorded data. The approach of data augmentation is considered to be useful for training end-to-end networks to reduce the over-fitting problem. The spacing between the electrodes is chosen to adapt more variation in height for different users.

3.2.3.2. Software implementation

The system operates at a sampling rate of 20 Hz, that corresponds to a time resolution of 50 ms. In Figure 3.36 such an input sample containing a repetitive movement is illustrated. Each samples has a time duration of 6 seconds window. The two visualizations are of the same signal. Line graphs is more intuitive to interpret when analyzing time series data. The image representation will be used by the convolutional neural network, which commonly operates on image data. To the right in Figure 3.36 the sensor identifiers are mapped to their location on the yoga mat.

The sensor numbering is not arbitrarily, since it serves to fit the kernel filter type of the convolutional neural network architecture. Different filter types such as locally connected or dilated filter would modelling different sensor correlations. Data appears to have a clear structure when performing repetitive movements in succession but is subject to high variance depending on the location on the mat and exact placement of the body parts. In capacitive proximity measure, we measure the modulation of the electric field. This change is caused by varying capacitive coupling between object interacting with the static electric field relative to the active face of the sensor. This electric signal is sampled by the analog-to-digital (ADC) converter.

The capacitive measurement is given by Equation (3.10).

$$y(t) = x_0 + n(t) + b(t) \tag{3.10}$$

The sensor value is defined by the static electric field generated by the sensor x_0 and the current environment coupling term $n(t)$ with a time dependence. The capacitive coupling between the human body toward the active face of the sensor is represented by the term $b(t)$. The measurements of the OCS board itself are stable and there is only minimal noise present results in a high signal-to-noise ratio (SNR).

An overall processing pipeline is visualized in Figure 3.37. In the following paragraphs, we will explain each pipeline step in detail.

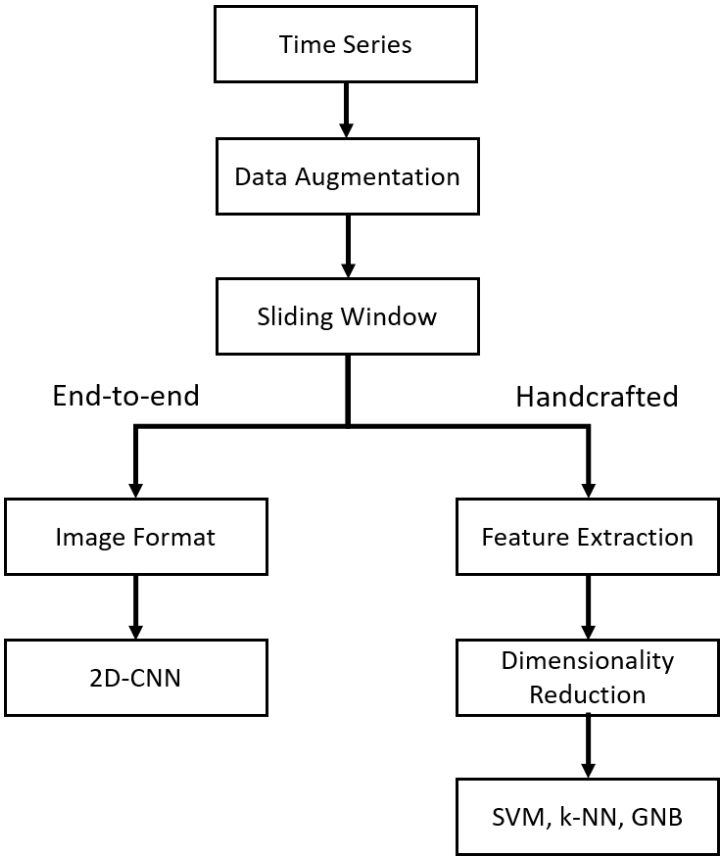


Figure 3.37.: the overall processing pipeline for the *ExerTrack* application is depicted

Data Preprocessing and Cleaning In this work, the data-driven approach is selected over model-based approach. The reason is that there exists a very strong inter-person variation due to the performance of the exercises and the location of execution on the sensing mat. Therefore, it is not feasible to learn a generalized classifier with a fixed model. The performance of the classification models based on data-driven approach is highly related to the underlying training data and its distribution. A clean data set with well labelled data and a broad distribution to cover all the variability can thus improve the performance and the generalization ability of the trained classification model.

The capacitive measurement has a baseline which is dependent on the environment it is placed in. The baseline could be different in different locations. This physical limitation of capacitive measurement made our data processing stage quite difficult. Therefore, we moved the data normalization stage inside the first layer of the neural network architecture to overcome the problem of varying baseline values. No further data cleaning is needed, since the signal-to-noise ratio from the human body is very high, i.e. $b(t) >> n(t)$, so that true modulation of the electric signal is dominating over noise.

Data Augmentation As we discovered early that there is a lot of intra- and inter-user variation in the collected data, we need to collect a lot of annotated data to reflect this distribution. Data acquisition process for activity recognition is however tedious and expensive. Thus only limited amount of samples are usually available to describe the true sample distribution. Therefore data augmentation is introduced to generate authentic and realistic synthetic signals. Data augmentation is a method to add prior knowledge into the data by transforming the data with methods that preserve the known label information [IYA16] and that model variations of the data we expect. As it is heavily used in deep learning methods for computer vision such as by Krizhevsky [KSH12] to reduce the generalization error of the model, it is not that easy to adopt it to time series.

We distinguish between **domain specific** and **general** augmentation methods. The former are bound to the system setup of *ExerTrack* and consist of interchanging the order of the incoming sensor readings to reflect the symmetrical sensor setup. The operations are as follows: The sensor values from one sensor are swapped with those of another, which is illustrated in Figure 3.38. If we interpret the incoming data as an image, they represent mirroring the data at the horizontal and vertical-axes. These operations mimic rotations of the user on the mat to some extent. We can argue that flipping the data is enough to model the user's orientation on the mat, as user's naturally orient themselves along the mat to perform exercises. By introducing these variations we expect to encode independence of the user's orientation on the mat.

The general augmentation methods include common methods such as *jittering*, *scaling*, *magnitude warping*, *cropping*, *permutating* and *time warping* [LMT16]. Due to specific sensor behaviours of our capacitive measurement, we can exclude jittering, permutating and cropping. Therefore the set of general manipulations is mainly focusing on magnitude and time warping, in order to reflect the realistic sensor data. Magnitude warping multiplies smoothly varying factors with the time series, that can be modeled with piece-wise cubic polynomial functions around 1 as suggested by Um [UPP*17]. We propose a similar approach to time warping, by interpreting the polynomials around 1 as time intervals: if we accumulate them, we receive new indices which can be used to resample the original data and interpret them as samples from the original indices again. This results in smooth warps along the time axis. We can model coarse variations with lower degree polynomials and fine variations with higher degree polynomials. This is illustrated in Figure 3.39, where a fine time warp is followed by a coarse time warp and subsequently a fine and coarse magnitude warp.

In our pipeline, the number of knots we choose for the piece-wise polynomials depends on the length of the time series. The transformations are applied to entire sets of exercises, which ensures continuity of the time series. This also means that we treat the created variations as synthetic data. When evaluating models later on,

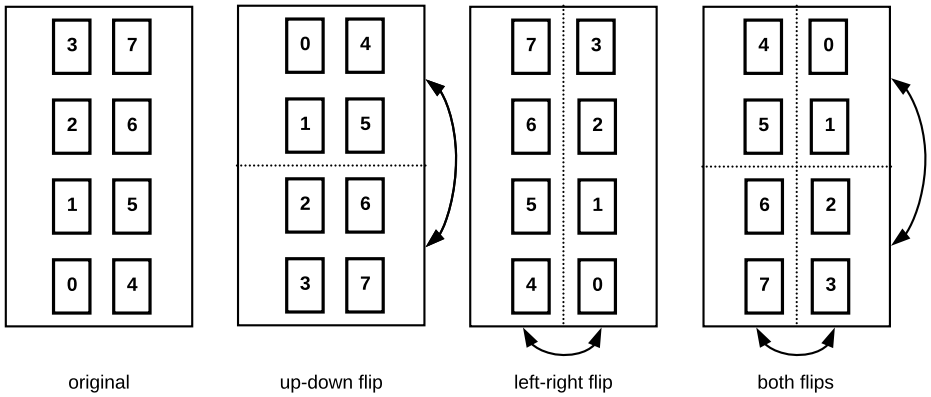


Figure 3.38.: Figure illustrates the domain augmentation method. The flipping of the sensor data corresponds to rotating the user on the mat.

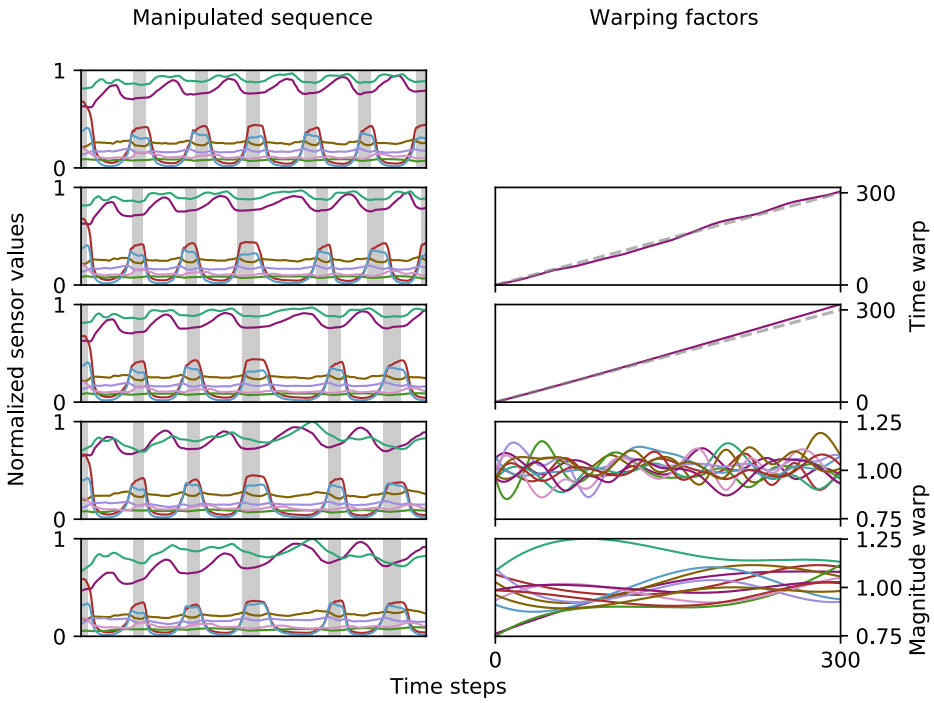


Figure 3.39.: The process of applying multiple label preserving warps to the same sequence. The gray areas in the left panel denote breaks between exercise repetitions. For *amplitude warp*, the grey dashed line indicates no warp $--$, the area above leads to stretching, while the area below squeezes the time series. By closely observing the modified time series in the left panel, we see a slightly stretched signal curvature where the curve is above the dashed gray line on the upper right Figure. For *time warp* a polynomial is accumulated and interpreted as time intervals to modify all sensor channels.

Augmentation method	Applied transformations	Data increase
domain	X_r with $r \in \{0,\dots,3\}$	1 : 4
general	$M_f(M_c(X,\sigma=0.2),\sigma=0.1)$	1 : 4
	$T_f(T_c(M_f(M_c(X,\sigma=0.1),\sigma=0.2),\sigma=0.2),\sigma=0.2)$	
	$T_f(T_c(M_f(M_c(X,\sigma=0.3),\sigma=0.6),\sigma=0.6),\sigma=0.6)$	
all	X_r with $r \in \{0,\dots,3\}$	1 : 8
	$T_f(T_c(M_f(M_c(X_r,\sigma=0.1),\sigma=0.2),\sigma=0.2),\sigma=0.2)$	
all extended	X_r with $r \in \{0,\dots,3\}$	1 : 16
	$M_f(M_c(X_r,\sigma=0.2),\sigma=0.1)$	
	$T_f(T_c(M_f(M_c(X_r,\sigma=0.1),\sigma=0.2),\sigma=0.2),\sigma=0.2)$	
	$T_f(T_c(M_f(M_c(X_r,\sigma=0.3),\sigma=0.6),\sigma=0.6),\sigma=0.6)$	

Table 3.20.: The tested data augmentation methods for which the evaluation is carried out. $r \in \{0,\dots,3\}$ denotes the original orientation and the three flips. M_f, M_c, T_f, T_c denote fine and coarse magnitude and time warps. The σ controls the variance of the polynomials around 1, i.e. higher values lead to higher factors and consequently greater distortions.

we only use this artificial data for training and not for testing. In Table 3.20 the data augmentation methods we evaluated for different classifiers are listed.

The **domain** augmentation method only consists of the previously discussed data flips and as such has no transformations of the sensor values. X_r with $r \in \{0,\dots,3\}$ denote the original orientation, left-right, up-down and both flips respectively. As a comparison, the **general** method only includes concatenated magnitude and time warps with an identical increase in data size.

The other methods combine the domain and general approaches by applying time and magnitude warps to each of the flips. The process of the **all** method is visualized for one orientation in Figure 3.39 and is repeated for each of the flips. It can be viewed as the more conservative approach, as it only applies one transformation to each of the flips. The **All extended** method concatenates the domain and general methods and as such leads to the highest data size increase.

The generated synthetic data is only used for extending the training data. Whenever we evaluate the predictive capabilities of the later described models we only evaluate against the original data, as to make sure that the models do not simply learn the augmentations themselves. This is important, since a negative learning is possible where the augmentations may *decrease* the performance of models [ZWY16], in cases they distort the data too much.

Processing Pipeline Activity recognition uses window segmentation of time series. The classification for each time window can be divided into distance-based, feature-based and end-to-end training. Distance based methods suffer from noise in the input data. Since there is high variation in the data even for the same user, depending on the exact location on the mat and the user's orientation, performing distance based classification on the raw sensor signal did not seem to be promising. Another drawback for distance based methods is the high evaluation time. As *ExerTrack* is meant to include an online application that either runs on a smartphone or a Raspberry Pi, this aspect is critical. If we e.g. examine dynamic time warping (DTW), the evaluation time for a univariate time

Feature	Parameters	Definition
mean	–	$\bar{x} = \frac{1}{n}\sum x$
variance	–	$\sigma^2 = \frac{1}{n}\sum(x-\bar{x})^2$
skewness	–	$G_1 = \frac{n}{(n-1)(n-2)}\sum_{i=1}^{n}\left(\frac{x_i-\bar{x}}{s}\right)^3$
kurtosis	–	$G_2 = \frac{n}{(n-1)(n-2)}\sum_{i=1}^{n}\left(\frac{x_i-\bar{x}}{s}\right)^4$
sample entropy	–	$-\log\frac{A}{B}$
absolute sum of changes	–	$\sum_{i=0}^{n-1}\lvert x_{i+1}-x_i\rvert$
autocorrelation	lag $l \in \{20,40,60\}$	$R(l) = \frac{1}{(n-l)\sigma^2}\sum_{t}^{n-l}(X_t-\mu)(X_{t+l}-\mu)$
number of peaks	$s \in \{\frac{20\cdot ws}{4}, \frac{20\cdot ws}{8}\}$	–
FFT coefficients	real, k $\in \{0\}$ absolute, k $\in \{0,5,15\}$ angle, k $\in \{15\}$	$A_k = \sum_{m=0}^{n-1} a_m \exp\{-2\pi i\frac{mk}{n}\}, k = 0,\ldots,n-1$

Table 3.21.: The extracted features for every sensor channel, resulting in 120 features (15x8) total for a time series segment.

series of length n is $O(n^2)$. This would have to be repeated for every sensor, which would drastically impair the reaction time of our final system. As such, we focused on a feature based approach that includes handcrafted and statistically relevant selected features in combination with conventional classifiers like support vector machine (SVM), k-nearest neighbors (k-NN) and Gaussian Naive Bayes (GNB) and as an alternative, automatic deep feature extraction with CNN.

First the time series data needs to be segmented to appropriate segments with local labels. According to a preliminary study, a time window of 6 second and an overlap of 50 % is selected to compensate for recognition accuracy and real-time response of the sensing system. Handcrafted feature selection is a tedious and time-consuming process. For this purpose Christ et al. [CBNKL18] proposed their Python library *tsfresh* (Time Series Feature Extraction on basis of Scalable Hypothesis tests). They streamline the computation of 794 time series features and parametrization. We used this framework to extract a subset of features based on the statistical relevance test and used them to train our classifiers. The list of extracted features, their formal definition and parameters are given in Table 3.21. All features (15) are extracted for each of the eight sensor channels, resulting in 120 features (15x8) in total for a given time series segment. Most of these features originated from the time domain, simply because their computation is fast. By applying the principle component analysis (PCA) method we further reduced the feature dimensions by 60% (from 120 to 74) while still conserving 98% of the information.

The suitability of conventional classifiers is compared further against deep learning approaches incorporating models using convolutional neural networks. The main difference is that the conventional models, such as GMM, k-NN and SVM, rely heavily on inductive biases, such as the prior expert knowledge of generating handcrafted features, while CNN are known for their ability to extract deep, meaningful representations from raw input samples. The process of extracting features becomes part of the optimization process for the whole model.

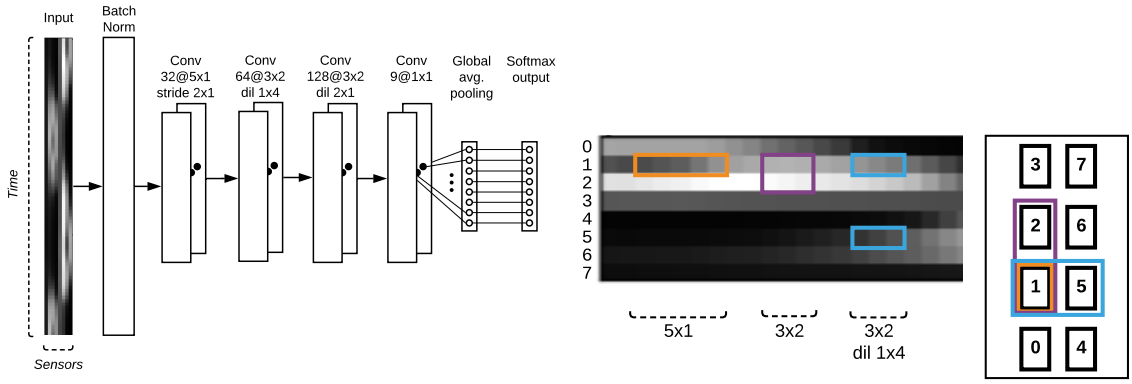


Figure 3.40.: The three different kernels used. Along the x-axis, features are extracted across time, and along the y-axis neighboring sensors are combined. These filters are the fundamental building blocks of a convolutional neural network. In the left panel, the sequential CNN structure is shown with the classification layer on top. The parameters of a kernel filter are commonly written as fs@$k_x \times k_y$ in the illustration of model architectures.

We investigated several models or hybrid models of convolutional neural network structures. We reason our choice on using convolutional neural network structure instead of sequence modelling structure such as the Long Short Term Memory (LSTM) network is due to the special sensor placement and it's ordering. Instead of interpreting the input sequence as a multivariate time sequence, we interpret it as a two-dimensional image, with spatial correlation across sensors over time. Literature reveals that different types of convolution filters are perfectly suitable for extracting local correlations. In our case, a convolution in the x-dimension represents a temporal convolution, i.e. finding patterns across time. In the y-dimension the values of the different sensor channels are convolved, allowing us to extract inter-channel features. We make use of dilation to encode relations between left-right aligned sensors, which let us to keep the sensor data in one dimension. The effect of different types of the convolution filters are illustrated in the right panel of Figure 3.40. If we encode the sensor values in a 2×4 fashion we would not need the dilation, but would introduce another dimension, increasing the complexity of managing the data.

The normalization over a batch is applied directly to the input during the training time. This step is crucial for our proposed application. While we could normalize the data beforehand, we argue that this process enables the network to deal with new data more flexible and robust. It does not depend on the global normalization of the training data as such. We experience heavy performance fluctuations when the test data had higher sensor readings than observed in the training data due to different baseline values according to various environmental coupling. Applying the input batch normalization on the training and test batch instead of normalizing the input data beforehand alleviated this problem. The output layer matches the number of classes plus a *none* class. A final global averaging pooling (GAP) layer is used to reduce the output parameters of the final convolutional layer and is directly connected to the softmax layer, which outputs the probability distribution over different classes.

In Figure 3.40 the sequential CNN architecture is depicted. After the input normalization, a temporal convolutional is applied with a 5×1 kernel. The stride of 2×1 results in halving the input along the x axis, to maintain the feature dimensions in the y axis in the deeper layers. As we want to give equal importance to features between neighboring electrode plates, we explore branching in our PBEF-CNN depicted in Figure 3.41(a), where we tried to fuse different types of convolution filters. Compared to the sequential model, we introduce another convolu-

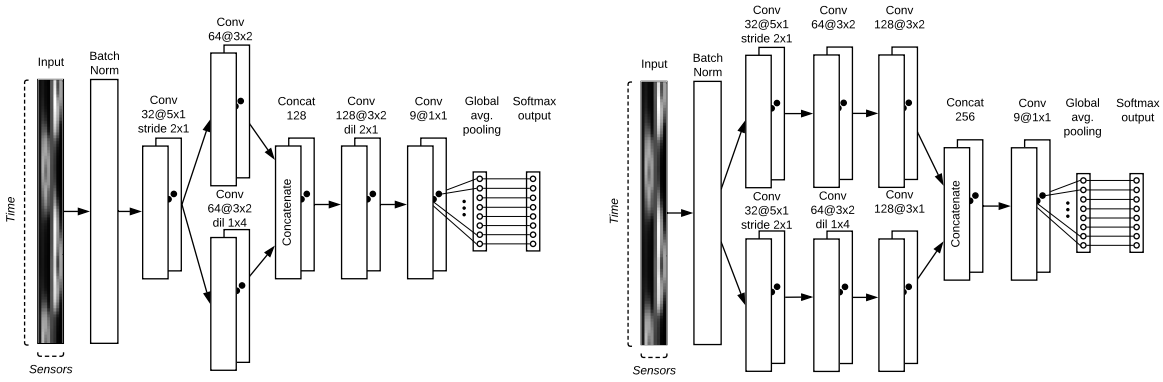


Figure 3.41.: (a) Early branch fusion PBEF-CNN model concatenating the structural information from spatially neighbouring electrode is shown. (b) This architecture investigates the performance of the model by first maintaining the separated low-level feature extraction and concatenate the classification at the final stage. We name this structure the PBLF-CNN.

tional block, but effectively do not increase the depth of the model, only the width. This architecture concatenates two different filter types to better catch the local correlations. Another tested model is the PBLF-CNN structure which implements late fusion as shown in Figure 3.41(b), where each branch consists of 3 convolutional blocks instead of one. In case of the late fusion, the low-level features are concatenated together in the final stage. This model extracts electrode neighborhood features at a later stage in comparison to the early fusion approach.

To address the challenge of model optimization, we use the Adam optimizer with a small learning rate of 0.0001. This is adjusted since we observed earlier that optimization was difficult and resulted in over-fitting to the train data very quickly. The models were implemented in Python with the *keras* [C*15] framework to easily create and test different architectures with its high-level API, using *tensorflow* [MAP*15] as backend. The models were tested and trained on a casual consumer PC with a *GeForce GTX 1060* graphic card.

Repetition Counting: The topic of finding exercise repetitions is commonly treated as another step after the classification task. We extend the method proposed by Sundholm et al. [SCZ*14] to fit our 8 channels sensor setup. The basic steps are described in the following order:

1. We create templates for individual exercises from the training data.
2. We construct the normalized cross-correlation of the template and the test data to find matches with the highest similarity measure.
3. We detect local maxima in the match score and count them as repetitions.

According to the data augmentation methods discussed in the previous paragraph, we can flip the data, resulting in 4 possible configurations $f \in \{0,\ldots,3\}$: original, left-right flip, up-down flip and both flips. This reflects all possible orientations a user can take above the sensing surface. To adapt step (1) to deal with these variations of exercises, we have to build a general template with re-oriented repetitions. While this alleviates the problem of averaging repetitions that vary greatly due to the orientation on the mat, it comes at the cost of having to flip and match the template three times to find repetitions later on.

With the calculated templates we can find matches by comparing them to an incoming signal using a similarity measure as described in step (2). For this purpose Sundholm [SCZ*14] used dynamic time warping (DTW). In our experiments, this proved to be too computationally expensive for real-time execution and did not yield any better results than using cross-correlation. Hence, the zero-normalized cross-correlation (ZNCC) between a template and a signal of the same length is the better measure of similarity in our proposed system and can be calculated as follows:

$$f(X,T) = \frac{1}{N} \cdot \frac{1}{\sigma_X \cdot \sigma_T} \cdot \sum_{n}^{N} (X_n - \overline{X}) \star (Y_n - \overline{Y}) \tag{3.11}$$

where $n = (0,\ldots,7)$ indicates the sensor channels. The benefit of using the zero normalized variant is that the results will always lie within the interval $[-1,1]$. We can use this to determine static thresholds to reject false peaks in step (3). The limitation of using the raw sensor data as input to our templates is that we need to know the orientation of the user and whether he is performing a left or right variation. We can simply flip the template in the remaining 3 orientations and calculate matches for each. The combined results from all possible orientations and the appropriate threshold value provide the final counting results.

3.2.3.3. Experiments and evaluation

The data is recorded in sets, i.e. all the exercises are executed consecutively. Transitions between exercises and breaks in between exercise repetitions are labeled as the *None* class. The length of sets ranges from 8 to 15 minutes, mostly depending on the length of breaks but also on the number of repetitions. For both the single and multi user data set the individual repetitions of each exercise are labeled manually by an external instructor during the recording.

The term *single-user* represents the person-dependent use-case where the model is optimized only for the single-user. The term *multi-user* represents the person-independent use-case where an inference model is trained on data from multiple participants. The collected data are further corrected afterwards with the help of a small interactive python tool and the video recordings. This is necessary as mistakes were made while labeling every exercise repetition. The labeling tool was much more helpful for labeling individual repetitions, because the exact start and end times were clearly visible in the sensor signals. The video recordings were used to ensure that the start and end times of the exercise periods matched. The classes regarding the exercises are fairly balanced. The relative distribution is roughly the same for the single- and multi-user data sets. The imbalanced data problem is handled in the model by introducing different weights for the classes depending on the class sample distribution.

The single-user data set consists of 12 sets performed by the same person. The sessions were manually labelled with a presenter stick during execution of the exercises. For leave-one-group-out cross validation, we assign sets that were performed on the same day to the same group, as the exercises are performed very similar in these cases. This separation results in eight groups. We split the data into six groups for training and two for testing. Hyperparamaters optimization is performed by applying leave one group out cross validation on the training set.

The multi-user data set consists of 17 sets by 9 different participants. All users except one contributed two sets of data, which were recorded on different days for each user. The reason to not record the sets in succession is that we aim to capture as much intra-user variability in the data as possible, reflecting the findings from the previously conducted single-user study. The participants were asked to start performing the exercises in the randomly predetermined order. The reason for the random order is two-fold: We want to ensure that we capture more variety in the transitions from one exercise to another and that trained models do not specifically learn the order of exercises. One of the goals during the data collection is to not heavily influence the participants in their exercise form. Only when they clearly misunderstood the task or had further questions, more detailed instructions

were given. For evaluation, we split the data into a train set consisting of 6 participants and a test set with the remaining 3 participants. For Hyperparameter estimation we perform leave-one-user-out Cross-Validation (CV) on the training set.

Evaluation on Classification Performance Cross-Validation is applied on the training data to optimize Hyperparameters for these models. It is however also interesting to aggregate the performance when training on the different folds, as we can estimate the generalization ability of the models. After this process, the model is fitted to the whole train set and evaluated against the test set. To ensure reproducible results, random seed for the training of the models and creation of augmented data were fixed. However, since the training of deep neural networks is computationally very complex and depends on nonlinearities it is not possible to reliably control the training. As such a minimalist difference in the model performance do not indicate which model is the better one. The three proposed architectures were trained in sequence, including the several proposed augmentation methods.

The box plots in Figure 3.42 denote the performance using leave-one-subject-out CV in the train set, which is also used for Hyperparameter optimization. Since we do the latter, it could be that the optimized parameters are overfitted to the training data and the models perform worse on the test set, which is why it makes sense to investigate. The performance on the completely unseen test set is indicated by the + scatter points. The best performance on the test set for the traditional models in the top row and the 2D-CNN based models below is highlighted for comparison. We use the weighted F1-score to compare the models. Since the class distribution is imbalanced with the *none* class dominating the other classes, the weighted F1-score is a better measure to compensate the ratio of recall and sensitivity in this multi-class scenario.

For k-NN, the optimal parameter is found to be $k = 2$ with a distance based weighting scheme, i.e. closer data points have a higher weighted vote for the assigned label. For the SVM the radial basis kernel(rbf) performed best, with an optimal penalty parameter $C = 11$ and $\gamma = 0.01$ (γ defines the influence of single training examples).

One of the first trends we can observe is introducing any method for data augmentation benefits the models, decreases the variance (indicated by the boxes) during leave-one-subject-out CV and usually increases the weighted F1-score on the test set. Especially the domain method, which only flips the data and does not manipulate the measured values directly, increases the performance and decreases the variance.

The depicted confusion matrices in Figure 3.43 show the correct predictions on the exercises. We compare the best conventional classifier with the best CNN model, according to their performance on the test sets, as highlighted in Figure 3.42. The k-NN appears to have the most trouble with the *None* class separation and with some exercises performed in the lying position. The PBLF model achieves better recognition rates for all exercises except for *quadruped*, which it confuses with *trunk rotation*.

The box plots for multi-user evaluation are illustrated in Figure 3.44 and the confusion matrices of the best performing models are shown in Figure 3.45. The conventional models perform a lot worse compared to the single-user evaluation, while the CNN architectures have only slightly worse performances. This confirms the presumption, that handcrafted features are often constrained and not robust enough to learn the inter-person variability. In the confusion matrix we can see that the SVM has trouble distinguishing exercises performed while lying such as for *bridge, segmental rotation* and *swim*, similar to the k-NN in the single user evaluation. The PBEF-CNN confuses barely any exercises with each other. It only performs slightly worse for the *push-up* exercise compared to the SVM but better for all others. As the separability of the *None* class is expected to be hard, it is acceptable to see confusion in this regard and we can conclude that the CNN outperforms the conventional classifiers by a great margin.

For the proposed CNN models, it is not clear which performs best. Due to the random nature of training deep neural networks, minor performance differences can be expected when training the same architecture more than

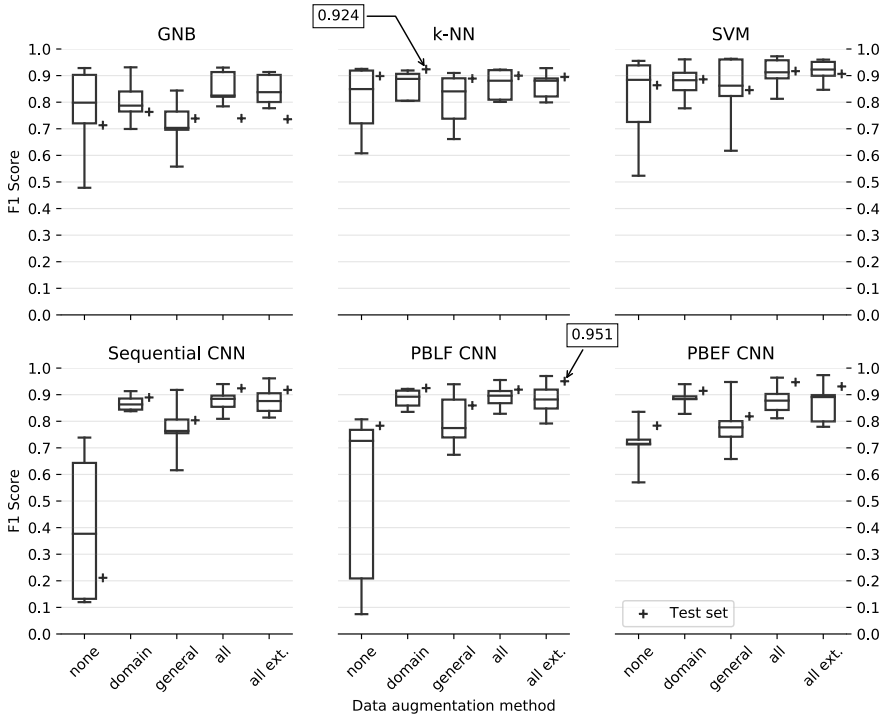


Figure 3.42.: Box plot for the person-dependent evaluation is shown in the left panel, conventional classifiers in the upper row and CNN models below. The different data augmentation variants are on the x-axis for each model. The + indicates the performance of our test set.

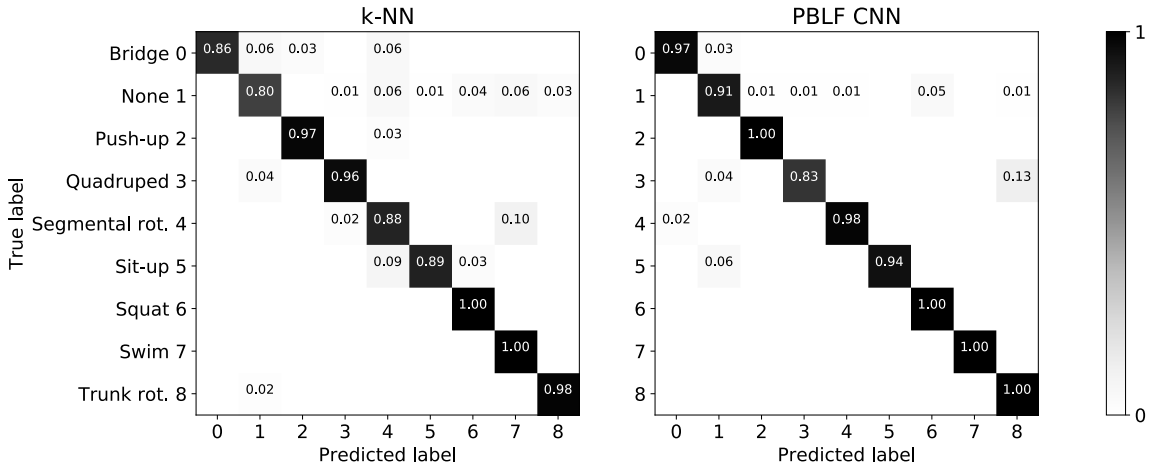


Figure 3.43.: Confusion matrices of the models with the best performance on the test sets for the single-user evaluation is shown.

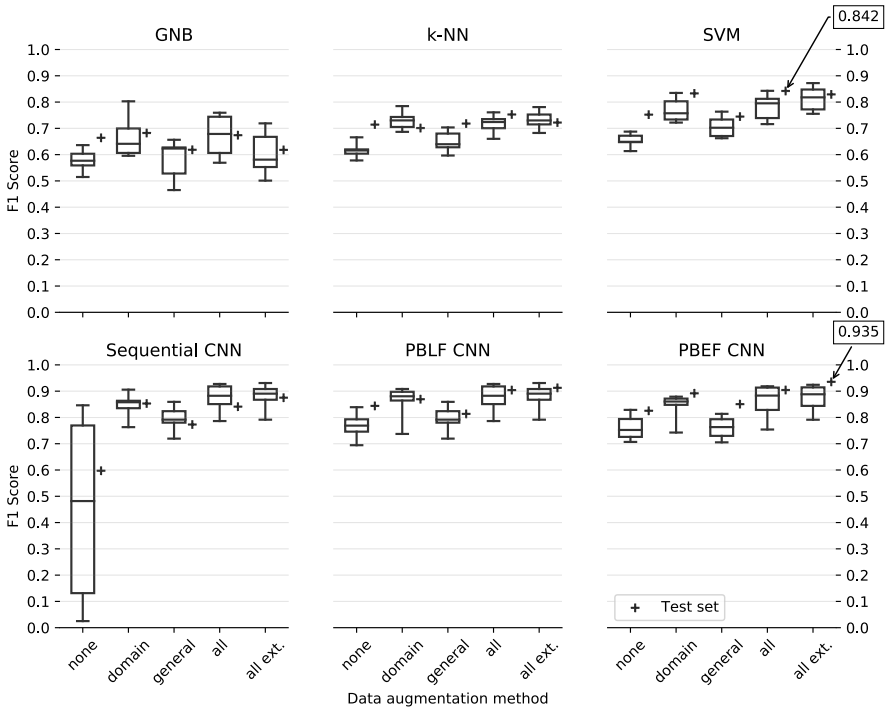


Figure 3.44.: Box plot for the person independent user evaluation is depicted in the right panel. The + indicates the performance of our test set.

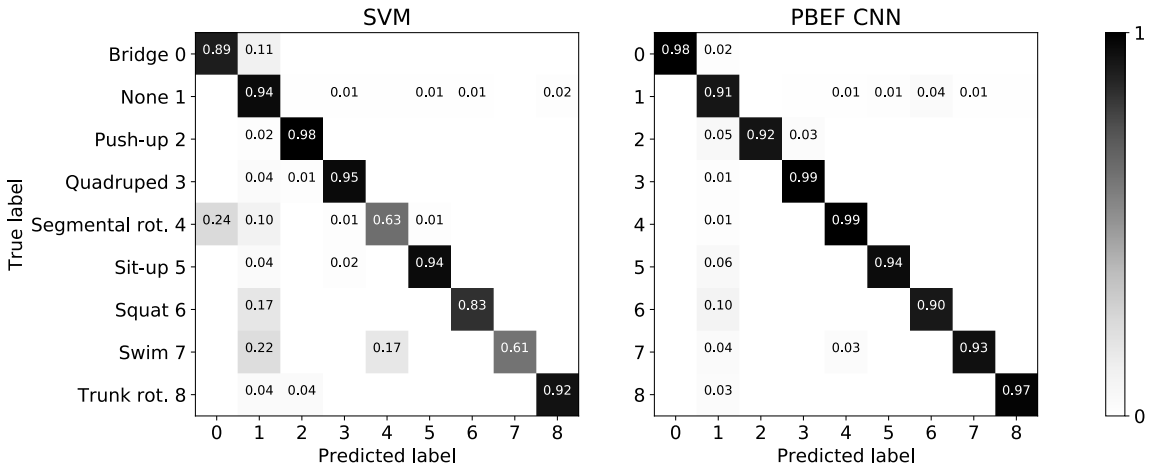


Figure 3.45.: Confusion matrices of the models with the performance on the test sets for the multi-user evaluation is depicted here.

Exercise	Threshold	Precision	Recall	F1 measure
Push-up	0.6	1.0	1.0	1.0
Sit-up	0.5	0.9884	1.0	0.9942
Quadruped	0.35	0.9839	0.9823	0.9831
Bridge	0.45	1.0	1.0	1.0
Trunk rot.	0.45	0.9638	0.8859	0.9232
Swim	0.5	0.9841	0.9725	0.9783
Squat	0.45	0.9224	0.5879	0.7181
Segmental rot.	0.5	0.9267	0.8932	0.9096
Average	–	0.9712	0.9152	0.9383

Table 3.22.: Selected thresholds and calculated metric for repetition counting in the single-user case.

once. However a general conclusion we can draw is that all CNN architectures profit from data augmentation, mostly so from the proposed domain method, that only includes flips of the input data. With this method a reliably low variance on the training data is achieved with leave-one-subject-out cross-validation.

The test set F1-score of the CNN models is consistently higher than the average trained CV F1-score, which is not always the case for the conventional classifiers. We can induce that using more real data benefits the deep learning models the most, as often recommended to increase confidence in models [Yos12]. Looking at the person independent case, the sequential CNN tends to have worse than average performance on the test set, as such we would rather use one of the branching models.

Evaluation on Repetition Counting In the single-user use-case, we perform leave-one-set/day-out CV to evaluate the template matching. The 12 sets by the same participant result in 8 iterations, as we do not use a holdout set. The 8 iterations correspond to the session on 8 different days of data collection. We create templates on the training set and try to match repetitions in the unseen test folds. We can view repetition counting as binary decision making – if we detect a peak within the labeled segment that indicates a repetition, we count it as a true positive. False positives are peaks that are in repetition breaks or in the transition phases (before starting/after finishing an exercise set) where no repetition actually exists.

We heuristically determined thresholds for the exercises to reject false peaks. Since we use normalized cross-correlation (NCC) a perfect match is 1. The thresholds are used to avoid false peaks which can occur in breaks between repetitions or in the transition from *None* to exercise and vice versa. A low threshold will lead to higher detection rates but might also introduce false positives, while a higher threshold will only detect exercises that are performed according to the template. The evaluation results can be seen from Table 3.22.

As can be observed from Table 3.22, class with clear time series structure, such as *push-up*, can use a relatively large threshold of 0.6 to achieve perfect counting accuracy. Classes with the left and right transitions require a smaller threshold $\leqslant 0.5$. These classes are for example *quadruped* and *trunk rotation*. The class *squat* has a relatively low recall score. This is because the exercise is far away from the sensing electrode and the maximum range of detection is around 15 cm close to the sensing electrode.

In case of the person independent counting case, we compare the evaluation metrics used by Sundholm [SCZ*14]. For each person, we checked which template from another person suits the best and count the repetition using the optimum template. Each positive count is for a peak in the normalized cross-correlation curve above certain threshold. Whereas we tried to find one general threshold for the same exercise throughout different participants to make the counting process more general and deterministic. In Figure 3.46 we calculated

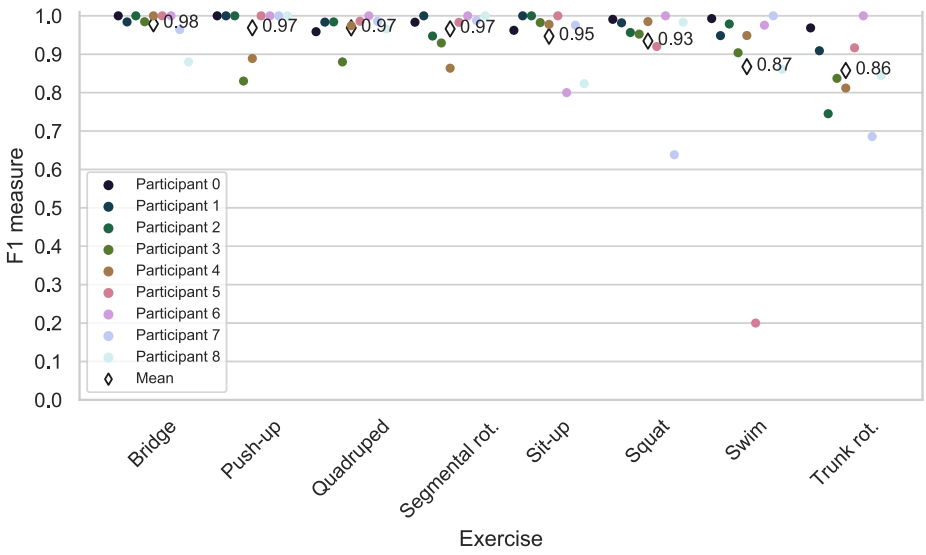


Figure 3.46.: Exercise repetition counting results for each participant and exercise using template selection in the multi user case.

the repetition by using leave-one participants out approach. The true positive and true negative matches will be counted to calculate the evaluation metric like F1-measure as depicted on the y-axis. Thus we achieved a user independent counting accuracy of 93.75 %. In case of exercises like e.g. *bridge, push-up* and *quadruped*, we achieve even higher counting accuracy compared to Sundholm [SCZ*14]. One possible explanation can be the nature of our sensing principle. As for proximity sensing a clear signal of the chest movement can still be detected in mid-air even without direct contact to the mat as opposed to pressure sensing.

3.2.3.4. Comparison to acceleration data from wearable

In order to demonstrate the advantage of our floor-based approach to single-worn acceleration data on the selected set of exercises, we simultaneously collected the sessions with users wearing a smartphone on the upper right arm. We make use of the 9-axis IMU data, including *linear acceleration, acceleration* and *gyroscope*. As the smartphone was attached to the upper right arm we assume difficulties when trying to disambiguate the exercises *bridge* and *segmental rotation*, where the user is lying on the floor and not actively moving the arms. All the other exercises include arm movements to some degree. This assumption has been confirmed by the raw acceleration data depicted in Figure 3.47.

A simple CNN with the same depth as the sequential CNN structure used in the capacitive sensor case is implemented here. The model only operates on the acceleration data as input. It similarly interprets the input as images, i.e. with dimensions of 120×9 using the same sampling rate. The achieved weighted F1-score for the acceleration data is only 84.61 %. Compared to the capacitive only solution, the weighted F1-score reduces around 6.74 percentage points. This outcome is expected, since the performance of a wearable is strongly correlated to the placement and should be task specifically optimized.

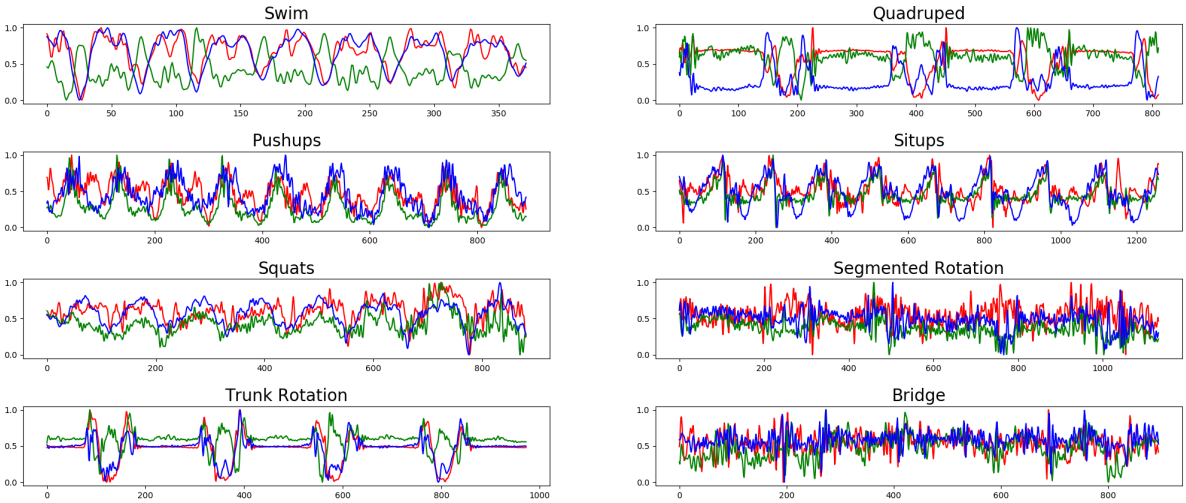


Figure 3.47.: Figure illustrates the acceleration data for the eight exercises from an entire sport session from one randomly selected participant. The red, blue and green curve represent the filtered and normalized acceleration in x, y, and z directions respectively. The x axis is the sample time dimension and y axis shows the normalized acceleration data. The class segmented rotation and bridge have the least modulation due to the dormant arm position.

other applications	bridge	push-up	quadruped	seg rot	sit-up	squat	swim	trunk rot
Exertrack	0.98	0.92	0.99	0.99	0.94	0.9	0.93	0.97
Ultrasound mobile	0.79	0.7	0.82	0.91	0.46	0.73	0.73	0.99

Table 3.23.: Accuracy depicted for person independent leave-one-out cross-validation. Value range lies between 0 and 1.

3.2.3.5. Comparison to ultrasonic sensing with mobile device

As both applications target at the same set of exercises, a fair comparison can be made across the systems. The following conclusions can be drawn. Exercise tracking with sparse sensor distribution performs better than using only one single sensor. Due to this reason, the yoga mat equipped with eight distributed capacitive proximity sensors outperforms in all exercise classes the mobile application with only one Doppler sensor as can be seen from Table 3.23. For accurate activity recognition with active capacitive measurement, a sampling rate of 20 Hz is sufficient. Ultrasonic sensing with a mobile device depends strongly on the back reflection and thus need the device to be close to the body for a good detection. With only one single sensing device, the placement is quite important to achieve a good accuracy. By leveraging Doppler measurement for a fine-grained body activity detection, we need to exploit the micro Doppler motions caused by secondary motions, such as legs or arms motion. Using the built-in hardware of a modern smartphone device is however more flexible than requiring extra hardware setup. Wearable applications exhibit good performance on a wide range of body-worn exercises, however it is strongly correlated to the sensor placement on body and is extremely task specific.

3.2.3.6. Discussion, limitation and conclusion

In this work, we equipped an off-the-shelf yoga mat with eight sparsely distributed capacitive proximity sensors to recognize eight workout activities. Despite the sparse sensor placement as opposed to other dense layouts, we trained several CNN structures to obtain good classification results. Regarding model selection, we have shown that CNN models can be applied in case only limited data are available to reach a weighted F1-scores of 95.1% on a user dependent study and 93.5% on a user independent study. They outperform conventional classifiers with handcrafted features from the time and frequency domain: a k-NN reached 92.4% in the former study and a SVM 84.2% in the latter. Thus we demonstrate the power of convolutional filters to catch locally correlated features across sensors by applying different filter variations.

Opposed to single wearable device, the collection of eight capacitive sensors increases the performance of recognition, especially for classes in which the acceleration data does not provide enough information. Although compared to pressure-based sensing, the proximity sensing further enhanced the ability of providing signals from remote objects without touching. This renders us the possibility of detecting more fine-grained activities.

Our proposed capacitive proximity sensing system however is by no means perfect. Regarding the hardware limitations, we need an active shield to reduce the environmental noise coupling to our sensing system from the ground surface. Further challenges lay in the nature of capacitive sensing, because the capacitive reading can be ambiguous. Further difficulties such as the problem of inter-person variability should be carefully addressed. This can be partly reduced by including synthetic data generation processes. However, other methods of adaptation has to be investigated to further improve the robustness and generalizability of the trained models. As resolving the issue of inter-person variability is essential for deploying the model for real-world use-case.

Different baselines with respect to the environmental noise also pose a problem to our proposed system. We target this challenge by using the normalization layer in the first stage of the neural network architecture, to make each input sample independent of the absolute values and leading the network to a faster convergence by restricting the input signal within a range of [-1,1]. The problem of intra-class variation is partly targeted by the pooling layer in the convolutional neural network. By reducing the size in the time dimension, the network can modulate the slow and fast variability of users performing the same activity. This can replace the need of using a time consuming dynamic time warping approach and thus makes it possible to build online applications.

Investigating different methods for creating synthetic data, we have shown that a system can heavily benefit if its sensor placement is chosen to allow for simple data augmentation methods (e.g. rotation). Applying the more general methods like magnitude and time warp for time series manipulation and concatenating them, have lead to the best performance for the proposed convolutional neural network models on both single and multi user data set. As the data acquisition process is costly and time consuming, this is an important aspect to consider when evaluating prototypes, as it increases the confidence in the evaluation of a system. This issue will be further considered in the Chapter 4.

3.3. Summary

This chapter addresses the research question 2 to 4, i.e. with respect to data acquisition, data processing and data modelling. In the following, I try to give some common and specialized strategies depending on the sensor application. These common strategies aim at providing useful guidelines for other practitioners in the research field of developing HAR applications with senors.

Research question 2 *What has to be considered for **data acquisition** with specific sensor technology?*

Sensors with on device processing unit and storage capability can be used directly to store data on the device itself. Sensor applications with self-designed hardware prototype often require to consider the capacity of the processing unit and the storage volumes. While the built-in storage on the mobile device is used to collect user data, we have to prepare a python-based recording script to store the data on a laptop PC connected to the OCS board to acquire the sensor values over the serial interface.

The system sampling frequency is restricted by the sensor hardware specifics. Because the audio sampling frequency of the smartphone is maximum at 44.1 kHz, we have to restrict the operating frequency of the continuous wave at 20 kHz. For self-designed prototypes, where the sampling frequency can be freely set, the practitioners can set the frequency accordingly to the task requirement. Detecting human motion with a sampling frequency of 20 Hz is sufficient such as for capacitive time series with smooth variations, when assuming the human locomotion for sport exercises is below 10 Hz.

In relation to sensor sensitivity, the experimental setup is to be finalized for data acquisition. While ultrasonic sensing can measure a distance up to 50 cm, capacitive proximity measurement requires a close distance of circa 15 cm in our proposed application to assure a correct measurement. The sensitivity varies depending on the sensor hardware specifications and thus limits the distance between the sensing unit towards the human body.

Research question 3 *What degree of **data processing** is sufficient without affecting the performance of the data modelling?*

Improving the signal-to-noise (SNR) ratio is correlated to the sensor proneness toward noise. Robust sensor hardware against noise impact does not require additional effort to improve the system SNR. Other sensors prone to environment noise require additional solutions to reduce the noise impact. This can be done either using hardware or software solutions. Hardware filters are leveraged to filter out known noise components. Software implementations aim at improving the signal quality by removing outliers or data impurities according to the knowledge of the data distribution.

Offline processing enables us to perform segmentation in regard to model performance. The segmentation of time series data often relate to the underlying task. The duration of the segmentation should at least include the entire activity, especially in case the activity classes have partial similarities. Improper segmentation leads the model to draw wrongly conclusions of the underlying data structure. General statistics on the activity classes provide a hint on the appropriate window length. In our proposed applications, a segmentation length of 6 s is selected according to the average performance time of the underlying sport activities with respect to model performance.

Normalization helps ranging the input signals to a pre-defined value range. This improves in general the comparability and reduces the difference in data. Spectrum normalization for the mobile device further mitigates the hardware specific diversity of the built-in microphone. Normalization on the capacitive time series similarly helps reducing the constant bias effect from the changing environment conditions. This step can be realized in two different ways, by either considering the normalization globally prior to modeling or is integrated in the model to be applied on each instance or batch. Relocating the normalization step to the model is more flexible towards real-world data, as the global normalization depends strongly on the given data distribution.

Feature extraction is performed prior to the modelling. For uni- or multivariate time series, this step includes generating features from time, frequency and/or time-frequency domain. A range of standard features, such as regarding the signal appearances or other statistical properties, can be extracted using common libraries. However, hand-designed features depend strongly on the inductive knowledge conditioned on the specific system. In addition, the pool of extracted features can contain strong correlations which do not provide any additional information to improve the context. Playing with these standard features however allows you to gain some first insight or intuitions about the data and the system. In case we have image data, computer vision techniques pro-

vide recent advances in automatic feature extractions to train end-to-end learning models. Knowledge extraction from image data can also benefit from transferring knowledge from domains with a huge backbone of similar experiences.

Research question 4 *Which architecture or model has to be used for* ***activity classification*** *in certain sensor application of HAR?*

Working with hand-designed features of limited dimensions and data, conventional machine learning approaches can be leveraged. Using machine learning approaches with restricted model capacity and pre-assumptions gives you a first impression about the separability of the underlying data. These machine learning models facilitate the ordering of the feature importance towards the decision making. Model capacity scales however with the amount of data. With the increasing amount of data, the performance of conventional machine learning approaches will meet the plateau at certain point, while the deep learning technique can still increase performance as they would strongly benefit from the increasing amount of training samples.

In comparison to using handcrafted features, end-to-end learning strategies mitigate the step of handcrafting features conditioned on the domain expert. The feature extraction and classification are combined into one optimization step. End-to-end learning models such as CNN models with variations of the kernel types are leveraged to model the cross-dependencies of the input data. While the standard kernel focuses on extracting the local neighborhood features, dilated kernel as a modified variation has a larger field of view to likewise consider features located further away.

Sequence models, such as bidirectional long short term memory networks are applied to catch the sequence information in data. The sequence information within the entire time segment is considered both in the forward and backward direction. This structure can improve the performance by combining the past and future information. This is especially useful addressing the activity classes that are partially similar to each other. In such a case, an overall consideration of the whole activity as a sequence is necessary.

Knowledge transfer ability of a pre-trained network with structural knowledge trained on a large backbone of normal images can be applied on other types of image data, such as the 2D Doppler spectrum as in our proposed use-case. This structure is called transfer learning. Training deep learning models with pretrained weights other than random initialization often leads to faster convergence. Including pretrained knowledge into your model is called warm start [LLP*14]. It not only leads to faster model convergence, but also motivates for a more efficient resource utilization.

4. Real-world data

Training on collected data under controlled environment can be finetuned to perfectly fit the underlying data distribution. However, we then often observe a performance drop from applying the trained model on data used under real-world scenarios. This can be due to various aspects. It can be due to simply overfitting the model on the training data. Or, it can be caused due to the missing diversity in the training data. Alternatively, it is due to the inherent difference in both dataset. In this chapter, I intend to answer the RQ5 *How to overcome the gap between constrained development data and the more complex real-world data with the scope on time series for HAR applications* by addressing the obstacles faced to successfully deploy a pre-trained model to adopt to real-world data.

Model performance relates to various aspects. It can relate to the model capacity, which is either too complex or too simple, leading to overfitting or under-fitting issues respectively. It can also be related to the given data, which might not be bias free. Poor data distract you from insightful analysis and cloud your judgement of the true data distribution. People often blame the model for the worse performance, however it is often the inherent difference in data distribution itself, which is making it difficult for the model to generalize. Typical issues are the following:

- not enough variability in the given data
- only limited amount of acquired data present
- inter-class and intra-class variability
- bias existent in the data

Not enough variability could make the model too simple, not able to model the true distribution in real-world scenarios. Data acquisition process for human activity data often proves to be difficult and the labeling process is not guaranteed to be correct and error free. Model design with limited data is difficult not to overfit. Finally, the human activity is complex in nature and thus causing a large intra-class variability and a small inter-class distance. Model should be able to cope with these listed issues in order to have an improved generalization ability.

All these factors make the given data the most critical factor to train a good model. Thus the focus of my research contribution in this chapter is to manipulate or understand the data to fit to various inference models and to increase the model generalization ability to adapt to changing real-world data. Both contributions are described as follows:

- Contribution 1: Data augmentation for capacitive time series: traditional versus generative models (Section 4.1)
- Contribution 2: Generalization of fitness exercise recognition from Doppler measurements by Domain-adaption and Few-Shot learning (Section 4.2)

In Section 4.1, I aim at improving the data diversity in the training phase by using time series augmentation. I compare diverse methods for synthesizing new samples such as signal appearance manipulation and probabilistic modelling. The trained inference model benefits from the increased data amount and the diversity added to improve the generalization ability of the model.

In Section 4.2, I aim at adopting the representation space from the training data and the real-world data by reducing their distance with a metric-based learning approach. In contrast to data augmentation, I explore methods which are adapted to cope with limited data amount. These common methods include domain adaptation and few-shot classification learning methods. I further categorize these methods with the requirements of retrain or without retrain the base inference model.

4.1. Data augmentation for time series

Large labeled quantities and diversities of training data are often needed for supervised, data-based modelling. Data distribution should cover a rich representation to support the generalizability of the trained end-to-end inference model. However, this is often hindered by limited labeled data and the expensive data collection process, especially for human activity recognition tasks, as these activity data are highly imbalanced. Extensive manual labeling is required. To mitigate this effort of massive data labeling, data augmentation is a widely used regularization method for deep learning, especially applied on image data to increase the classification accuracy. But it is less researched for time series. In this section, we investigate the data augmentation technique on continuous capacitive time series with the example on exercise recognition. We intend to show that the traditional data augmentation can enrich the source distribution and thus make the trained inference model more generalized. The generative probabilistic models such as variational autoencoder or conditional variational autoencoder can further reduce the variance on the target data. This section is mainly based on our work published in [FKK20a].

4.1.1. Introduction

In order to train data-driven inference models for especially deep learning networks, large quantity of supervised training data are beneficial [ZBH*17]. In the domain of computer vision, billions of labelled images can be easily acquired from public available databases. Thus enabling deep learning methods to build successful applications [LBH15] such as image recognition and object localization. However, the data acquisition for human activity data is expensive and the labeling process is error prone. Compared to images, the shortage of labelled data for time series from sensory input is an impeding factor to train generalized inference models. Several minority activities such as *falling* or *running* are much more difficult to acquire compared to activities such as *walking* or *sitting*. Therefore, data augmentation should be used to generate synthetic data, especially for such minority classes. The objective of such generative model is to learn the probabilistic distribution of the hidden representation of the activity classes and then generate more new samples similar to the true distribution of the real samples. According to Goodfellow [IYA16], generative model is a method to introduce prior knowledge into the data by transforming the data with methods that preserve the known label information.

Another inherent problem of human activity recognition is its high degree of complexity and uncertainty. It is difficult to build a robust and generalized activity recognition model which suits all individuals. This problem is called user-diversity. The goal is thus to find a robust feature representation to model this diversity with limited data. In this section, we address the issue of user-diversity by using data augmentation methods for time series. We call the data of one user group source data, whereas our target data is another unseen user group. By applying data augmentation, we aim at making the distribution of our source distribution wide enough to cover the target distribution and thus make the inference model more powerful, performing on the unseen test data. We distinguish between traditional and generative models. The former includes domain specific methods or geometric modifying methods on the raw sensory input data. We claim that the source distribution will be enhanced by traditional data augmentation techniques on the raw signal space. The generative models such as

variational autoencoder (VAE) or conditional variational autoencoder (cVAE) can be used to further decrease the variance on the target data.

Therefore, our objectives of using data augmentation on time series are as follows:

- to increase the generalization ability of the inference model, while preserving the temporal dynamics
- better adaptation to unseen data by manipulating the source distribution, i.e. to enrich the source distribution
- to learn a probabilistic distribution of the hidden representation of input data by leveraging generative models
- to use ensemble of generative models to generate more variations of output samples

This Section is organized as follows: we first describe related approaches in the research topic of data augmentation. We then detail various proposed methods separated into classical traditional data augmentation methods and generative data augmentation methods. An enhancement of using ensemble method to build generative models is also provided. Evaluation with real data from the experiment collected in our previous application is conducted. A final discussion on the challenges and limitations of the data augmentation methods on time series are included with some closing remarks.

General researches showed that deep learning applications [KSH12] can benefit from data augmentation techniques. Various research papers such as [FFW*18] showed that data augmentation can improve the generalization capability of deep neural networks, especially in many computer vision tasks such as image recognition and object localization. In the following, I first present several data augmentation techniques on various aspects used in the common literature.

Data augmentation incorporated in the model itself On the model itself, Dropout is a common technique to increase the model capacity to adapt to new data samples. Bouthillier et al. [BKVM15] proposed to use dropout as a kind of data augmentation technique in the input space without domain knowledge. Adding noise at the output by applying dropout can be viewed implicitly as applying transformation in the input space. Thus projecting the dropout noise back into the input space is viewed as generating augmented versions of the training data. They showed that training a deterministic network with dropout yields similar results as training the network on the augmented samples with the benefit of avoid adding significant computational cost.

Data augmentation on the data side It is used as another regularization technique to train robust classifiers which can handle various different shapes of the same input data. The goal is to train classifiers that can handle slight modification of the input without affecting the output of the classifiers. By adding noise, rotating, scaling or cropping an image of a bird would not change the label information. The appearance of it still resembles an image of a bird, unless the object of interest is occluded after the applied transformations. Therefore, data augmentation for images are more intuitive than for time series data. For time series data, this is not as easy. While certain classes carry clear structure in the signal, it is hard to say which transformations will change the original data too much and thus make the label information incorrect. This could affect the inference model in a negative way.

Data augmentation on image data Generative adversarial networks (GANs) are commonly used to generate synthetic images from unsupervised training data. Frid-Adar et al. [FAKA*18] compared traditional versus GANs to enlarge the training set with the intention to increase the classification accuracy. The traditional methods were assumed to increase the data size, while GAN network is supposed to increase the diversity to the original data pool. Both methods were applied on a strongly limited amount of computed tomography (CT) images of

182 liver lesions. The classification accuracy of only using classical data augmentation achieved 78 %, while synthetic data augmentation with GAN further increased the classification accuracy to 85.7 %.

To generate samples not randomly, but from predetermined classes, we need to further incorporate conditions in the synthetic generating process. Those approaches are for example conditional generative adversarial networks (cGANs) or conditional variational autoencoders (cVAEs). Mishra et al. [MKRMM18] used cVAE to generate images for zero shot learning tasks conditioned on attribute vectors. Evaluating on four benchmark dataset, they demonstrated that their model outperformed the state of the art. Wang et al. [WLZ*18] leveraged conditional GANs to synthesize high-resolution photo-realistic images conditioned on semantic labels.

Data augmentation on time series Current researches on data augmentation for time series are most common in feature spaces. Oversampling or under-sampling are popular cases to balance the class distribution. However the former could easily lead to overfitting since it only duplicate data without modulation and the later will change the input distribution and thus lead to loss of useful information. Synthetic Minority Over-sampling Technique (SMOTE) introduced in year 2002 [CBHK02] for synthetic feature generation with k-nearest neighbours are used to deal with the aforementioned problems by carefully generating new synthetic samples instead of copying.

SMOTE is commonly applied directly in the feature space. According to Wong et al. [WGSM16] the classification results can benefit more if the data augmentation technique is applied directly in the original space instead of feature space. The author evaluated on two data augmentation techniques, by creating additional samples with data warping method in the data space and synthetic oversampling in features space. They evaluated the performance on standard MNIST handwritten digit dataset over a range of classifiers. Those include a convolutional backpropagation-trained neural network, a convolutional support vector machine and a convolutional extreme learning machine classifier. Both data augmentation techniques yield an increase in classifier performance. However, it is shown that it is more beneficial to perform the data augmentation in data space, as long as label preserving transformations are possible.

Alzantot el al. [ACS17] used a generative model of stacked LSTM cells combined with a Mixture Density Network (MDN) to synthesize sensory data. To introduce more variability on the generated synthetic data, the MDN network was applied on top of the LSTM architecture. Their objective was to generate fake samples to deceive the discriminator and making it unable to distinguish between generated synthetic samples and the real samples. But their main objective was to protect user privacy data by replacing the real data set through a synthetic set and not focus on the task of improving the classification accuracy as targeted in our use-case.

Our goal is to investigate data augmentation techniques on time series in a specific domain with respect to classification performance. The objective of using data augmentation for time series is to be considered as a regularization technique to build more generalized inference model with the ability to cope on unseen target data. Adapting to real-world data, the data augmentation is used once to train the classifier. The trained model is used in inference time on new data samples. In order to preserve the hidden inherent data distribution, we choose variational autoencoder and the conditional variational autoencoder structure as our generative model. The hidden probability distribution of the input data will be approximated by a known function. This approximate function is sampled to reconstruct new samples which are subject to the same inherent data distribution. The conventional augmentation techniques mainly target at the manipulation of the signal appearance.

4.1.2. Our proposed methods of Data augmentation on capacitive time series

In this work, we use data labeled and collected from the application introduced in Chapter 3.2.3. *ExerTrack* is a capacitive proximity sensing system that is able to detect non-stationary exercises. It consists of a regular exercise mat enhanced with eight capacitive proximity sensors attached underneath to perform exercises recognition. The

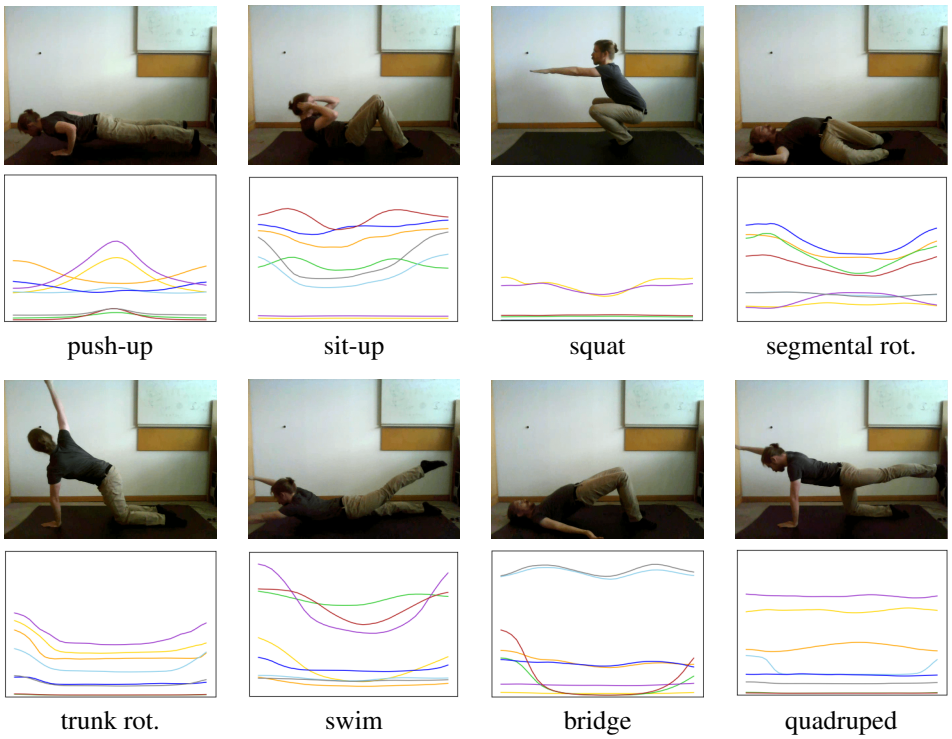


Figure 4.1.: Figure shows the sport exercises we collected and the capacitive time series with respect to each exercise is shown below [LJ18].

sensing principle is based on active capacitive sensing where the sensor generates a low-frequent quasi-static electric field. The presence of a non-conductive body, such as the human body will affect this static electric field. This modulation in electric field strength caused by body movement with respect to the sensing electrode is measured by the sensing entity.

All exercises corresponding to its raw sensory input time series are depicted in Figure 4.1. These exercises are periodic in movement that causes periodic modulation in capacitive time series. Opposed to pressure-based sensing technology, no direct contact to the sensing mat is required in our proposed system. It can measure up to an interaction distance of 15 cm. This property of remote sensing enables us to catch more fine-grained body actions close to the sensing device even without contact. The sampling frequency of the operating system is 20 Hz. This operation frequency is fast enough to cover the targeted velocity of these eight chosen activities. The time window for each activity is selected to 6 s optimized to the classification performance from our previous investigation. This results in a windowed input sample of the dimension 120x8, meaning 20x6s = 120 discrete time samples for all eight sensor channels.

Capacitive time series defines sensor data collected from low-frequent capacitive measurement. In contrast to acceleration data, capacitive data are less noisy and there occurs less abrupt changes in the time-series. Signal curvature is more smooth compared to high frequent signals (e.g. acoustic signals). Thus this type of signal is easy to model, predict, and has less anomaly. Capacitive technology has been widely used in the field of human activity recognition, such as posture detection [VKMV11] for healthcare applications, appliances for

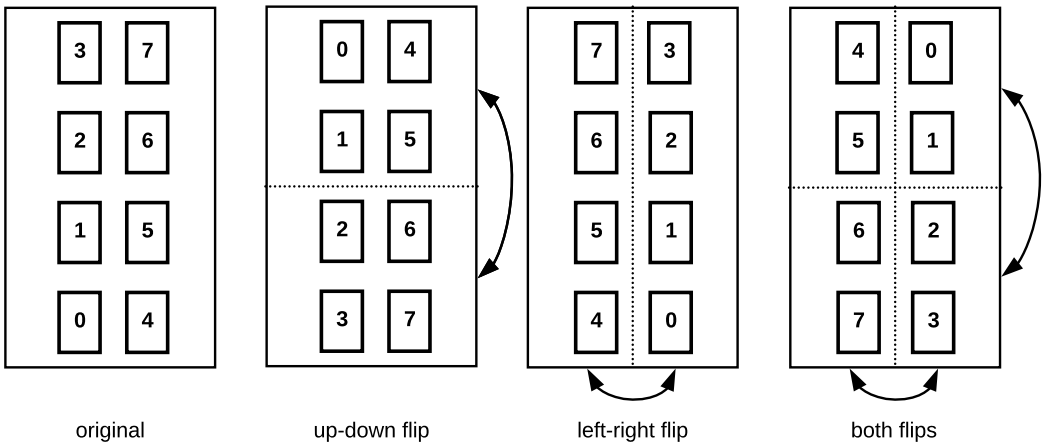


Figure 4.2.: Illustration of flipping the sensor data, which corresponds to rotating the user on the mat [LJ18].

smart home [BHW11, GHP11] and wearable applications in form of flexible textiles [SNR*15, Teo13]. However, publicly available databases for capacitive data with respect to human activity recognition is still a niche. Most prominent works collected their own database for prototyping without sharing them with the research community. Thus, our work intends to present first results on the possibility of using simple data augmentation techniques on capacitive time series to increase the diversity of input data in the data space. This method should not only generalize to capacitive time series, but also generalize for other types of signals whose property resembles the capacitive measurements with respect to the signal smoothness and differential in time.

4.1.2.1. Traditional data augmentation

Due to the symmetric system layout of 2x4 sensor placement, one simple method is a domain specific technique. As users are not constrained in which direction the exercises should be performed, the original windowed time series data can be flipped in several ways as depicted in Figure 4.2. The amount is increased by a factor of four by applying this method.

Other traditional methods implement magnitude and/or time warps. These methods directly modulate the appearance of the raw input time signal. Magnitude warping multiplies smoothly varying factors with the time series, which can be modeled with piece-wise cubic polynomial functions around 1 as proposed by [UPP*17]. We extended this approach to time warping, interpreting the polynomials around 1 as time intervals: if we accumulate them, we receive new indices which can be used to resample the original data and interpret them as samples from the original indices again. This results in smooth warps along the time axis. We can model coarse variations with lower degree polynomials and vice verse. This method is depicted in Figure 4.3, where a fine time warp is followed by a coarse time warp and subsequently a fine and coarse magnitude warp. The gray dashed line indicates no warp –, where the area above leads to stretching and the area below associates with them the squeezing of the time series. These manipulation is applied to the whole input time series before the windowing process to preserve the smoothness of the overall signal. In Table 4.1 the traditional augmentation methods we evaluate for different classifiers are listed. The σ controls the variance of the polynomials around 1, i.e. higher values lead to higher factors and consequently greater distortions.

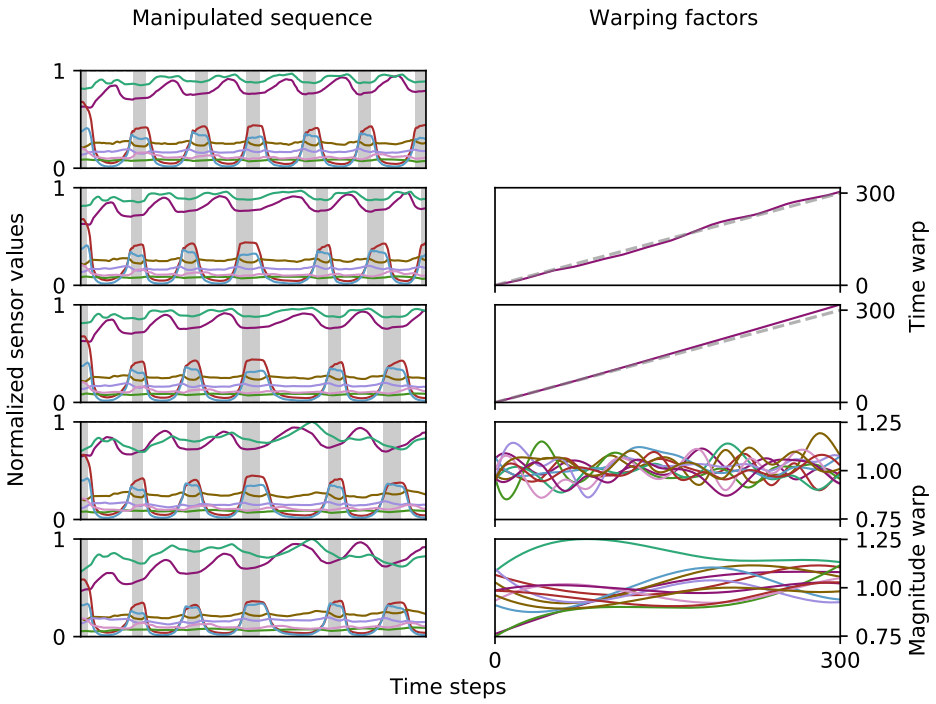


Figure 4.3.: The process of applying multiple label preserving warps to the same sequence is depicted. The gray areas on the left denote breaks between exercise repetitions. For *time warp* a polynomial is accumulated and interpreted as time intervals to modify all sensor channels [LJ18]. The dashed gray line indicates no modification.

Augmentation method	Applied transformations	Data increase
domain	X_r with $r \in \{0,\dots,3\}$	1 : 4
general	$M_f(M_c(X,\sigma=0.2),\sigma=0.1)$	1 : 4
	$T_f(T_c(M_f(M_c(X,\sigma=0.1),\sigma=0.2),\sigma=0.2),\sigma=0.2)$	
	$T_f(T_c(M_f(M_c(X,\sigma=0.3),\sigma=0.6),\sigma=0.6),\sigma=0.6)$	

Table 4.1.: The tested data augmentation methods for which the evaluation will be carried out. $r \in \{0,\dots,3\}$ denotes the original orientation and the three flips. M_f, M_c, T_f, T_c represent fine and coarse magnitude and time warps. The σ controls the variance of the polynomials around 1 [LJ18].

To visualize the training sample distribution, ISOMAP [BS02] is used to reduced the high dimensional time series representations to two dimensional feature embedding space. Isomap is a non-linear dimensionality reduction method for computing a quasi-isometric, low-dimensional embedding for a set of high-dimensional data points. It learns the internal manifold of the high dimensional input data based on a rough estimate of each data point's geodesic distance to the others. We trained the ISOMAP on the original time series and mapped the augmented time series in to the same feature space. In Figure 4.4 the training data distribution is shown. Different colors represent the individual participants. The '+' indicates the original samples and '*' indicates the augmented samples. Clearly visible is the effect of domain specific augmentation, as it mirrors the original distribution in various ways. Also the general warping shows impressively the effect of enhancing the input distribution.

4.1.2.2. Generative models

The generative approaches are the second large category of data augmentation methodologies, we will consider in this thesis. Generative models indicate the synthetic generation of new samples based on statistical models of the underlying data generation process. We have adopted three different methods by using: variational autoencoder, conditional variational autoencoder and the weighted combination version of both autoencoder.

Generative method 1 One of the generative approaches we used is the method called variational autoencoder (VAE) proposed by Kingma et al. [KW13] and Rezende et al. [RMW14]. It is a powerful generative model used to generate fake images [PGH*16] or purely synthetic music [TY17] relying on the modeled statistical knowledge about the input space. It can be viewed from the perspective of unsupervised representation learning. By gaining a deep knowledge about the input space even without labels, we will be able to alter, or explore variations on already existing data and in a desired, specific way.

VAE tries to build a model of the input space $p(\vec{X})$ using the observations of input training data $\vec{X} = \{\vec{x_1}, ..., \vec{x_N}\}$. Here, the input data are the windowed time series, while the observations are the discrete sampled values. The objective of the VAE is to generate realistic sensor time series similar to samples from the true hidden distribution of the input space. Since the true hidden distribution $p(x)$ is intractable, the task of VAE is to approximate a simple known distribution $q(z)$ which can approximate the true distribution. Then sample from this approximation $q(z)$, we can reconstruct realistic new samples subject to the true sample distribution. The easiest way to do this is by applying the maximum log-likelihood estimation method (MLE). A latent variable $\vec{z}$ is introduced to marginalize over to get the same distribution as shown in Equation (4.1). The symbol θ represents a possible set of the networks hyper-parameters.

$$\mathcal{L} = \sum_{i=1}^{N} log\, p_\theta(\vec{x}_i) = \sum_{i=1}^{N} log \int p_\theta(\vec{x}_i, \vec{z}) dz \tag{4.1}$$

In general the dimension of the latent variable $\vec{z}$ is smaller than the input variable $\vec{x}$ space. The process of compressing the information from the input variable space $\vec{x}$ to latent representation $\vec{z}$ is called the encoder stage. We can further express the distribution over the input space by applying the Bayes theorem, with the Equation (4.2).

$$p_\theta(\vec{x}) = \int p_\theta(\vec{x}, \vec{z}) dz = \int p_\theta(\vec{x}) p_\theta(\vec{x}|\vec{z}) dz \tag{4.2}$$

This allows us to generate new samples conditioned on the latent distribution z. If we have the prior probability $p_\theta(\vec{z})$, we can now sample value $\vec{z}$ from this distribution in order to generate the value $\vec{x}$ from the posterior

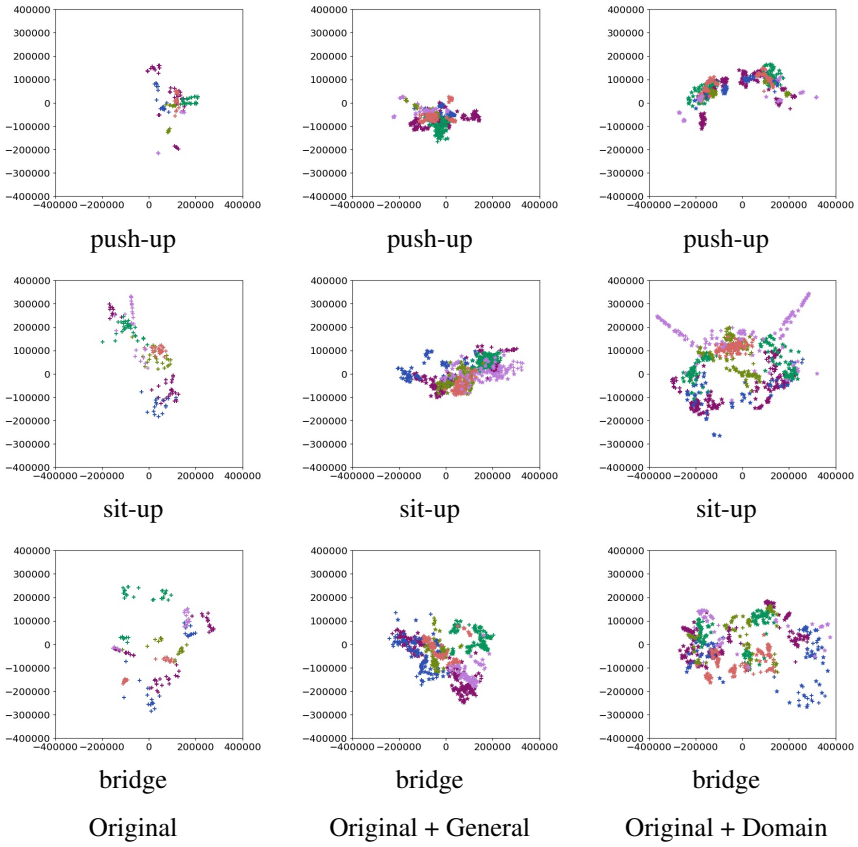


Figure 4.4.: Two dimensional feature space is visualized for three of the eight sample exercises. The color represents the 6 individuals used for training. The + is the original sample and * indicates the augmented sample. Domain specific data augmentation reflects the assumption of adding flip information by mirroring the sample distribution. Magnitude and Time warp further introduced a more diverse feature distribution.

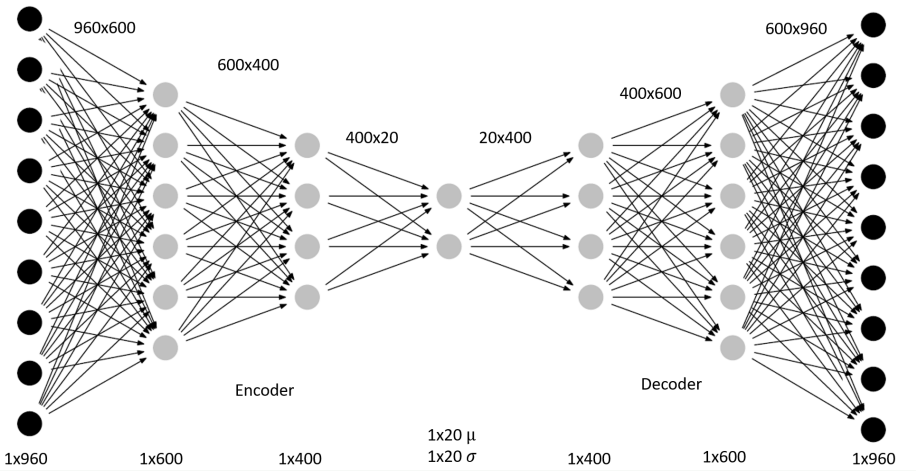


Figure 4.5.: Pipeline of the variational autoencoder using two successive fully connected (fc) layers is depicted here. Each hidden layer in the encoder and decoder stage is consisted of a batch-normalization, relu activation, and a dropout layer to regularize the VAE network. The latent space is learned in the encoder stage.

probability $p_\theta(\vec{x}|\vec{z})$ by using a generator network. This is called the decoder stage. To optimize the encoder and decoder stage, we have to minimize the following objective in Equation (4.3).

$$p_\theta(\vec{z}|\vec{x}) = \frac{p_\theta(\vec{x}|\vec{z})p_\theta(\vec{z})}{p_\theta(\vec{x})} \tag{4.3}$$

The loss function of the original VAE is given in Equation (4.4). It consists of two separate parts, the first term presents the log-likelihood function which refers to the reconstruction error. The second term is to make the approximated posterior distribution $q(\vec{z}|\vec{x})$ and the model prior $p_\theta(\vec{z})$ more similar to each other. It is called the Kullback-Leibler divergence [KL51].

$$L(q) = \mathbb{E}_{\vec{z}\sim q(\vec{z}|\vec{x})} log\, p_\theta(\vec{x}|\vec{z}) - D_{KL}(q(\vec{z}|\vec{x})||p_\theta(\vec{z})) \tag{4.4}$$

As we have very limited training data for deep learning approach, we used a very shallow model with limited capacity. The architecture is illustrated in Figure 4.5. We reshape the input shape of 120x8 of a windowed input sample to a sequence of the dimension 1x960. In the encoder stage, the input dimension will be reduced in two successive fully connected (fc) layers, each accompanied with a batch-normalization layer, reLu activation layer and one dropout layer. The dropout rate of 0.3 is used to regularize the network from overfitting.

After the latent variable space is learned, the output is generated using the decoder stage. Similar to the encoder stage, we successively increase the dimension of the latent variable z by using two fc layers to generate new input samples. This version of VAE works very fast and takes the overall time series interconnections into consideration. Samples of the generated output of $\vec{x}$ are depicted in Figure 4.6. The architecture should be adapted to the input space. In case of a more complex input domain, a deeper network structure is to be applied. In order to enrich the input distribution and not just memorizing, we further manipulated the latent space variable before

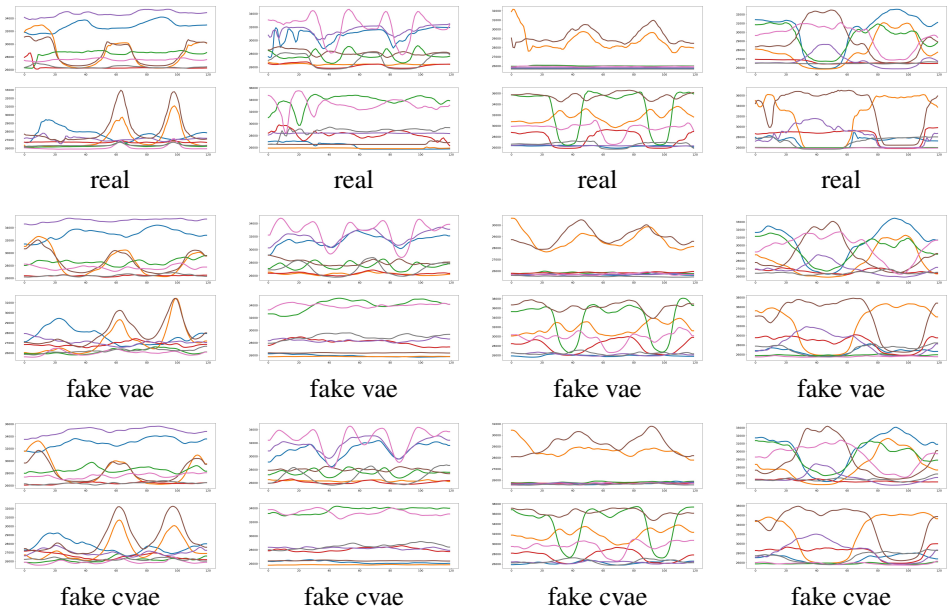


Figure 4.6.: Sample outputs from VAE and CVAE compared to the real output for the 8 individual exercise classes are shown. The first two rows depict the original samples.

the output generation. Assuming samples surrounding the true sample have similar structures and labels, we added a small normalized Gaussian perturbation on to the multidimensional hidden variable z. Other possibility is to set a probabilistic value p to select which dimension should be perturbed by the normalized Gaussian noise. Since the output of the VAE is *shaky*, a smoothing filter in form of a moving average filter with the kernel size of 3 samples was applied to the reconstruction. The width of the smoothing filter is yet another hyper-parameter that can be adjusted according to the generated output data.

Generative method 2 In case of conditional variational autoencoder (cVAE), we further included the class label as conditional constraints in the encoder and decoder stage. The cAVE is an extension of VAE. In case of VAE, we trained the generative model in an unsupervised way, therefore we have no control over the data generation process. Opposed to VAE, this method allows us to generate samples conditioned on the class label. The objective function for VAE is given in Equation (4.5).

$$\begin{aligned} &logP(X) - D_{KL}[Q(z|X)||P(z|X)] = \\ &E[logP(X|z)] - D_{KL}[Q(z|X)||P(z)] \end{aligned} \tag{4.5}$$

The encoder $Q(z|X)$ is only using training data X without its label information. The decoder $P(X|z)$ is also dependant solely on the latent variable z. As a solution, cVAE proposed by Sohn et al. [SLY15] enables to generate new samples with specific attributes. This allows the generative model to generate samples conditioned on specific attributes such as the class labels. The objective function extended to include the class label of the

input sample as given in Equation (4.6).

$$\begin{aligned} logP(X|c) - D_{KL}[Q(z|X,c)||P(z|X,c)] = \\ E[logP(X|z,c)] - D_{KL}[Q(z|X,c)||P(z|c)] \end{aligned} \tag{4.6}$$

In this way, for each given class label $c = y$, the network is trying to model its own distribution $Q(z|X, c = y)$. Therefore it is a good way to incorporate labels to VAE, if available.

The model structure is similar to VAE illustrated in Figure 4.5, with the exception, that we now also include the class label c as one-hot vector to the encoder and decoder stage. We merge the one-hot vector with another small dense layer to previous input vector x. The latent vector size z remains the same as in case of VAE. But we also merge the class label c to the latent variable z in the reconstruction stage for the new generated output $\hat{x}$. Samples of the generated example outputs of the cVAE model is shown in Figure 4.6. On the first two rows, the original signal of the 8 different classes is depicted. Compared to the original data, the next two sets are both generated from VAE and cVAE models with the same original input with minor modification on the latent space before the reconstruction stage.

In this network architecture, we try to use Maximum Mean Discrepancy (MMD) [TSS16] loss instead of the Kullback-Leibler divergence. The reason why MMD is favoured to KL divergence is that the MMD is a non-parametric measure. Thus no estimate of parameter from the underlying distribution is required. We can work on dataset directly. MMD represents the distance between two probability distributions by a distance between the means of the embeddings and its variance in the feature space. The equation of measuring the MMD distance is given by Equation (4.7),

$$\begin{aligned} F = \mathbb{E}_{p(z),p(z')}[k(z,z')] + \mathbb{E}_{q(z),q(z')}[k(z,z')] \\ -2\mathbb{E}_{p(z),q(z')}[k(z,z')] \end{aligned} \tag{4.7}$$

and thus the overall loss function for a MMD-VAE transferred into Equation (4.8) from the original VAE loss function in Equation (4.4).

$$\begin{aligned} \mathcal{L}_{\text{MMD}-\text{VAE}} = \text{MMD}(q_\phi(z)\|p(z)) + \\ \mathbb{E}_{p_{data}(x)}\mathbb{E}_{q_\phi(z|x)}\left[\log p_\theta(x|z)\right] \end{aligned} \tag{4.8}$$

We identify that for training each individual model, the performance with MMD loss is higher compared to KL divergence loss. According to [ZSE17], the MMD-VAE is always preferred especially for small dataset. Because for small dataset, KL divergence tends to over-fit and generates poor samples, while MMD loss still can generate reasonable samples.

4.1.2.3. Ensemble of generative models

In order to wisely composite both characteristics of VAE and cVAE models, we further explore the architecture by combined training of both generative models together in one network. We extended the objective to optimize both loss functions from both models. The combined loss function is presented in Equation 4.9.

$$L(q) = L_{rec,CVAE} + L_{KL,CVAE} + L_{rec,VAE} + L_{KL,VAE} \tag{4.9}$$

The loss function consists of the reconstruction loss of both generative networks and the KL divergence loss to minimize the distance between the probabilistic distribution of the generative output and true input distribution. We trained for 1000 epochs, with the objective to reduce this combined loss function. For combined training of both generative models, the KL divergence converges better than MMD and therefore is a more suitable choice

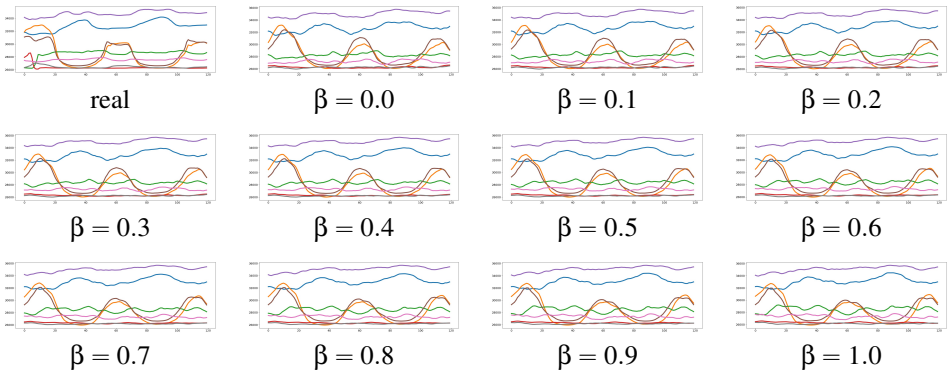


Figure 4.7.: Sample outputs of the ensemble generative model for the combination with different weight parameter are depicted.

here. The network structure of VAE and cVAE are introduced in the previous section. They have separated structures, but the back-propagation is performed on the ensemble loss function.

The new generated output samples are given by the weighted combination of both generative outputs as in Equation (4.10). This weight factor can easily affect the generated signal output by the underlying model. Varying this factor can shift the importance to one of both underlying models.

$$\hat{x}_{rec} = \beta \cdot x_{rec,VAE} + (1-\beta) \cdot x_{rec,CVAE} \tag{4.10}$$

In Figure 4.7, we observe the generated samples with varying weight factor from 0 to 1.0 with a step size of 0.1 each. This effect corresponds to a shift from favoring the VAE at the beginning to favoring the CVAE model at the end. Thus ensemble method can produce more variability and are intended to further increase the classification performance. It also induces more variance in the inference model. Therefore the enhanced performance comes with a price of increased variance. Proper design is required to train the ensemble model and to combine the generative outputs. Data science practitioners should weigh the advantages and disadvantages before applying the ensemble model.

4.1.2.4. The evaluation classifier architecture

The evaluation architecture used for the classification task is a simple convolutional neural network (CNN) model. We are training inference model from an end-to-end network to avoid handcrafted feature generation. The process of handcrafting features and forming discriminative representations to increase the classification performance is strongly related to domain expert knowledge. Thus the model generalization ability is not always guaranteed and is thus out of scope. Here, we intend to prove the assumption that generating new time series by using data augmentation techniques in data space, can make the inference model perform better on unseen input samples. Convolution filters have proven its superior ability in extracting useful features from each sensor channels. By modifying the convolution filter types such as adding dilation, we are further able to capture features from across the sensor channels.

The architecture for the evaluation network can be seen in Figure 4.8. To avoid overfitting, the L2 regularization is used for the weights in each convolutional layer. Batch-normalization and Dropout Layer with a dropout rate of 0.2 are added after each convolutional layer. The output is a global average pooling (GAP) layer with

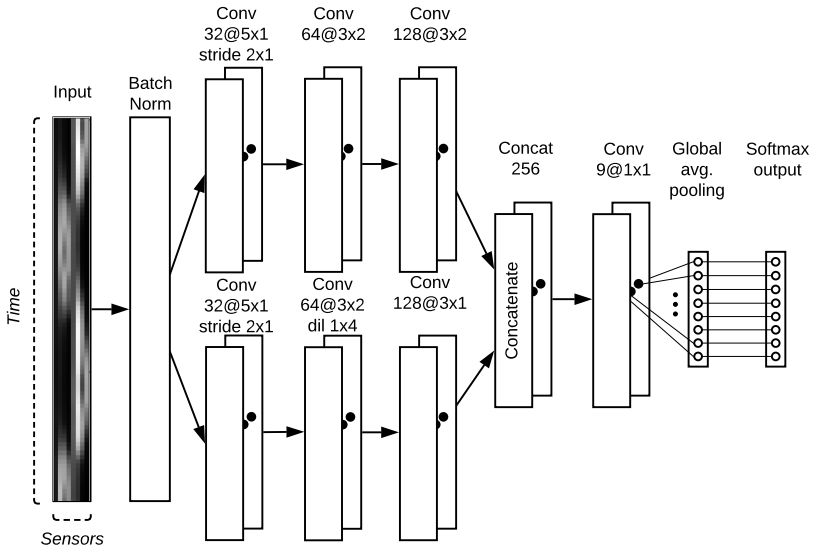


Figure 4.8.: CNN model used to perform the classification task in the evaluation.

Setting	Training	Test
1	Original (Baseline)	Original
2	Original + Domain	Original
3	Original + General	Original
4	Original + Domain + VAE	Original
5	Original + Domain + CVAE	Original
6	Original + Domain + Ensemble	Original

Table 4.2.: Table contains the different setups for the conducted experiments.

the number of outputs equal to the classes of activity plus the none class. The none class includes all transitions between useful activity and none activity. The model uses a fixed batch size of 200 and a fixed number of training epochs of 40 for all different experimental setups. The learning rate is fixed at 0.001.

4.1.3. Experiments and evaluation

Evaluating the proposed methods, we collected data from 9 individuals, each containing two sessions of the 8 different exercises. We randomly selected all sessions from 6 individuals as training set and keep the sessions of the other 3 individuals as holdout set to evaluate the performance of the trained inference model. We used 5-fold cross validation to fine-tune the inference model. The holdout test set is used to measure the performance of the trained model and its variance on this unseen data. The 6 different setups are listed in Table 4.2.

In the second setting, we only use original dataset to include its three flip versions as given in Table 3.20. This results in a four times augmentation from the original data amount. As in the third setting, the original dataset

Setup	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	σ^2
Original (Baseline)	55.8 %	55.9 %	59.5 %	65.6 %	60.3 %	3.59
Original + Domain	86.3 %	**87.0** %	85.6 %	82.0 %	86.9 %	1.84
Original + General	84.2 %	82.5 %	84.8 %	**86.7** %	86.4 %	1.53
Original + Domain + VAE	87.8 %	**88.6** %	88.4 %	87.1 %	86.9 %	**0.68**
Original + Domain + CVAE	87.2 %	86.8 %	86.8 %	**88.7** %	88.4 %	**0.81**
$(\beta = 0.5)CVAE + (1 - (\beta = 0.5))VAE$	**88.7** %	**89.0** %	**89.9** %	83.0 %	85.0 %	2.65

Table 4.3.: Table contains the F1-score on the unseen test data from the inference models trained using the 5-fold cross validation on the training samples with different settings.

is warped in time and magnitude with a coarse and dense warp as shown in Table 3.20 and thus also results in a four times increase. The overall class distribution for the samples remains the same as in the original case.

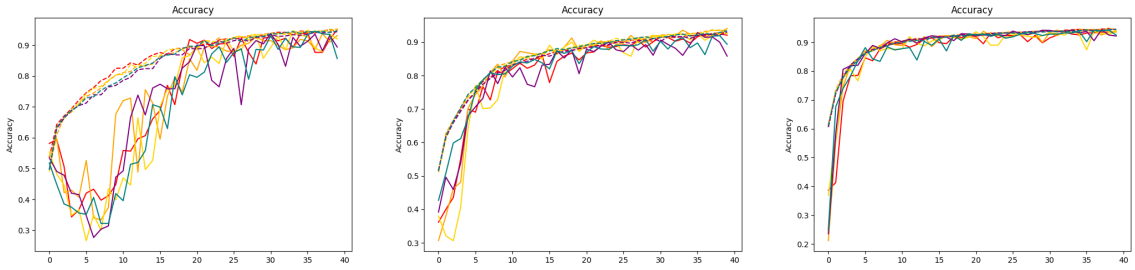
For the generative model, we used the original dataset including the domain specific augmented samples to train the generative models. In case of VAE, the training is conducted in an unsupervised fashion without the labels of the training data. Whereas in case of conditioned VAE, the training data label is used to condition the generated output signal. The label is included both in the encoder and decoder stage. In this way the generative output is conditioned on the input label. To train the ensemble model, we simultaneously combine the training of both generative models in parallel branches. The training objective is thus to reduce the reconstruction error on both models simultaneously to minimize the Equation (4.9). The final output generation is concatenated with a weighting factor $\beta = 0.5$ as stated in Equation (4.10). This weight factor β controls how strongly each individual generative model will affect the generated output signal.

Now, in Table 4.3 we list the evaluation results on the holdout test set using the trained inference models using 5-fold cross validation on the training data. We have chosen the weighted F-Measure (F1 score) to illustrate the classifier performance since the training dataset is not fully balanced. This evaluation measure provides a geometric mean of sensitivity and specificity for each class. As expected, the weighted F1 score on the holdout set is increased around 25 to 30 percentage points by using data augmentation in general. Both warping in the *general* case and integrating additional domain expert knowledge due to the symmetrical system design, have increased the performance by more than 20 percentage points compared to baseline. Further it is interesting to note, that the expected model variance on the holdout set can actually be reduced by using further generative models such as VAE or cVAE models. The highest performance is shown by leveraging ensemble of generative models. This could be explained that each individual generative model captures something unique of the hidden representation space and are able to generate multi-variate output spaces in combination.

Similar results can be seen on the performance curve for the training set. The weighted F1 score on the training and validation set are visualized in the Figure 4.9 and 4.10. We observe a strong increase in the accuracy, the more diversity in data the model was exposed to during the training phase.

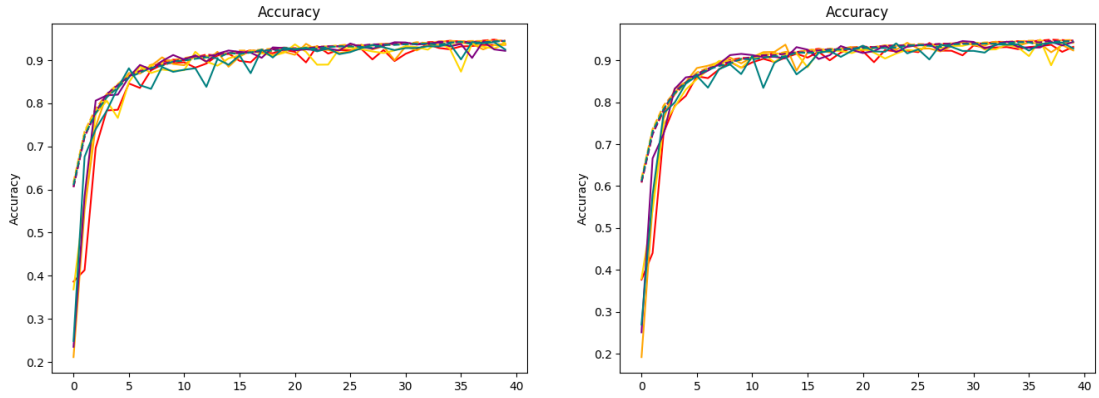
4.1.4. Discussion, limitation and conclusion

Labelled data of time series from sensory input is often limited and the acquisition process is tedious and expensive. Especially for the task of human activity recognition, not every activity can be acquired equally. For example it is way difficult to collect as much real falling activities as general activities, such as walking or running. In addition, time sequence can not be easily shuffled, rotated or permuted without destroying the sequence information. That is why not much research has been conducted on time series augmentation than on data aug-



Original (Baseline) Original + General Original + Domain

Figure 4.9.: Figure depicts the accuracy curve of 5-fold cross-validation results from different settings. The dashed curve represents the training accuracy, while the solid curve represents the test accuracy. Increased performance can be seen on models with data augmentation techniques.



Original + Domain + VAE Original + Domain + CVAE

Figure 4.10.: Figure depicts the accuracy curve of 5-fold cross-validation results from different settings. The dashed curve represents the training accuracy, while the solid curve represents the test accuracy. Variance is reduced for the training samples with respect to original data set or data set augmented with traditional data augmentation only.

mentation for images. However, several researches already proved that deep learning can clearly benefit from large quantity and diversity of labeled training data. Therefore it is of interest to address the problem of time series augmentation.

We identify that special care has to be taken into account by designing network architectures for augmenting time series data. A distorted bird image can much likely still to be recognised as a bird by the human visual than a largely distorted time signal. The degree of distortion is therefore an important hyperparameter to optimize for augmenting time series to still preserve the original label. Similarly it is the case for the generative models. Generating new realistic samples, we reconstruct slightly modified samples in the learned hidden feature space. How large can the manipulation of the new sample be apart from the mapped original sample in the latent space, without the risk of generating a wrong sample? These parameters should be carefully selected.

Therefore, a careful design is required to apply data augmentation for time series. It could have a negative effect, if the traditional data manipulation is too strong or the generative model is not properly designed. In general, we observe a positive effect by applying traditional data augmentation on time series to enhance the input data distribution and thus increase the model generalization ability on unseen holdout set. The generative models can further reduce the variance on the holdout set. The setup of the ensemble model by wisely fusing both generative models can further increase the performance on the holdout set. The fusion process is controlled by the weight parameter β to determine the importance of each individual generative model contributing to the generated sample.

Conclusion Data augmentation on images is often used to improve the performance for deep learning models. Data augmentation on images is easy, as the manipulation on the pixel values does not change the appearance of the object too much. However, data augmentation on time series is not that simple, as the manipulation would distort the signal too much. In this work, we proposed several methods used to augment capacitive time series in order to improve the model generalization ability and preserve temporal dynamics. The methods used in this work can be divided into two main categories. The first one is the traditional way of time series augmentation. It contains the domain approach and the general approaches. The second one is the generative way of time series augmentation. It contains three alternative methods to synthesize new samples in an unsupervised, a supervised and a combined way.

Domain methods include only basic transformations such as flipping and reordering the time series. General approaches include amplitude and time warp methods. Time warp leads to stretched and broadened signal appearances. Magnitude warp leads to signal appearance deformations. Generative approaches aim at generating samples from learning the underlying data distributions. We applied an unsupervised learning method by using the variational autoencoder (VAE) and a supervised learning method by using the conditional variational autoencoder (cVAE). VAE learns the underlying data distribution without the label information, while the cVAE learns to generate samples conditioned on the underlying labels of the input sample. The combined generative method fuses the unsupervised and the supervised structure to generate new samples. This method aims at combining individual advantages of both generative models. By changing the weight of the fusion parameter, you can manually choose to put more importance on one of both generative models.

Based on the evaluation results, we observe that by only using the traditional domain augmentation method an improvement of 30 percentage points is achieved. Similar improvement can be observed by using the general approaches. These are unsupervised methods by manipulating the appearance of the raw input signal in relation to self-selected control parameters. The generative methods further improve the variance on the cross-validation test results. The fusion of generative approaches achieved the highest performance by combining both domain and generative methods. These methods benefit from modeling the hidden statistics of the original time series to generate new samples.

Data augmentation on time series is only valid if both data distributions are similar. Large difference in data distributions (development set and test set) makes this technique inappropriate to synthesize new samples. However, this is often the case for real-world datasets. Addressing this challenge, in the next section, we are considering methods, which can be applied on data with dissimilar or disjoint data distributions.

4.2. Other ways to increase the model generalization ability to real-world data

In Section 3.1.7, a mobile application was developed using an unmodified commercial off-the-shelf smartphones to recognize whole-body exercises. The working principle was based on the ultrasound Doppler sensing with the device built-in hardware. Applying such a lab-environment trained model on realistic application variations cause a significant drop in performance, and thus decimate its applicability. This is induced by the variations induced by the user, environment, and device variations in realistic scenarios. Such scenarios are often more complex and diverse, which can be challenging to anticipate in the initial training data. To study and overcome this issue, this section presents a new database with controlled and uncontrolled subsets of eight different exercises. We also propose two concepts to utilize small adaptation data to successfully improve model generalization in an uncontrolled environment, increasing the recognition accuracy by two to six folds compared to the baseline for different users. This section is mainly focused on our accepted work in [FDKK20a].

4.2.1. Introduction

Human motion is highly complex and possess a high degree of freedom. This is expressed with the term user-diversity. Making learning a generalized model for all possible variations of a human motion very challenging. Training a model on limited amount of individuals under constrained environment often leads to a large performance drop when applying the model on individuals/environments disjoint from the training data. The large drop in performance originates from the large variations between the controlled training data and the real-world application scenarios. This can be caused by the diversity and complexity of the users actions, the device hardware or other environmental variations. However, all possible reasons lead to a degradation in the usability of the proposed application. One possible solution is to reduce the inherent difference between the development dataset and the real-world dataset by making the development data resemble the real-world data. However, due to the diversity in the real-world applications, there is no generalized model that is applicable in all possible situations.

This work addresses sport exercise recognition from a stationary smartphone using the Doppler measurement as proposed in Section 3.1.7. We propose a set of methods to improve the generalization of a pre-trained model (trained on controlled data) to scenarios containing a combination of unseen environments, individuals, and devices. To achieve this, we propose and investigate two concepts, along with a clear baseline that demonstrate the generalization problem. We have developed a mobile application aims at collecting data that is used to deal with this challenge. Our application uses the built-in hardware of a commercial off-the-shelf smartphone to measure whole-body exercise activities. The main contribution of this work is grouped as follows:

- A novel database for investigating micro-Doppler motion in relation to whole-body exercise data with built-in smartphone hardware. The database contains sessions in controlled environment, as well as a disjoint subset containing variations of environments, individuals, and devices.
- Propose and adapt two concepts (with variations) to improve the recognition generalization. These concepts are based on domain adaptions, as well as few-shot learning. Both concepts proved to enhance the generalizability on data variations in comparison to a clear baseline.

The structure of this section is organized as follows: we first provide current researches on the topic of finetuning approaches with the focus on model generalization to fit new data and categorize these approaches under two main categories (retrain required and not). We then introduce one of the main contributions by presenting our collected database. We further propose the baseline model and several new approaches targeting our problem, under two main concepts. Evaluation results of our proposed individual approaches using our proposed database is introduced along the baseline. We further discuss the advantages and disadvantages of certain methods and provide some guidelines in design choice for such an application.

The problem of data diversity Andrew Ng stated in [Ng17], that the most frequent fail of inference model in reality is the inherent difference between your development set and test set. If the development set can not represent the true distribution of the test set, the trained model will not generalize well on real-world data.

To overcome this problem, effort can be put into developing great development/training databases. Though, it is impossible to make the development set identical to the test set (application scenario), due to the large variety and complexity in human activities. Common methods to adapt the pre-trained model on individual new data samples without enforcing much restrictions on the development set is desirable. We distinguish between two main categories: with and without retraining the base model to adapt to new data.

Retraining of the Base Model Domain adaptation builds on transferring knowledge from similar domains to cope with unknown target domains, where the target domain samples do not require to have labels. This method is especially useful, when you do not have enough annotated datasets for the particular problem at hand. The goal is to extract knowledge from related, known datasets, and use this knowledge to learn the new task at hand.

Wang [WZCH18] benefited from using similar labeled source domain data to annotate the target domain which has only a few or even none labels. They evaluated their approach on acceleration dataset from different body positions as different domains. To alleviate the problem of negative knowledge transfer, they proposed a measure to choose the *right* source domain with respect to the target domain. They proposed an unsupervised source selection algorithm to select the right source domain which shows the most similar properties related to the target domain. Afterwards, an effective transfer neural network is used to perform knowledge transfer from the selected source domain to the unknown target domain.

Khan and Roy et al. [KRM18] proposed a CNN based transductive transfer learning model to adapt action recognition classifiers trained in one context to be applicable to a different contextual domain. The limitation is that the set of activities being monitored is the same in both context domains, as they are transferring knowledge from individual convolutional layers. Evaluated on their acquired smartphone and smartwatch acceleration dataset on 15 users and 8 activities, they demonstrated the ability of their proposed methods on transferring knowledge from smartphone to smartwatch domain and vice versa. By incorporating a small amount of target labels, they were able to further increase the performance.

These methods pose less constraints on the target domain, however are difficult to train, as the knowledge transfer is solely based on the source domains. Thus the choice of the appropriate source domain is critical for the performance of the inference model on the unknown target domain. A retraining is required to relate the source domain to the new target domain due to knowledge transfer.

No Retraining of the Base Model Few-shot learning is currently an active research field in machine learning. The ability of deep neural networks to learn complex correlations and patterns from a vast dataset is proven. However, current deep learning approaches suffer from the problem of poor sample efficiency. To make a model learn on a new class, sufficient amount of samples from this class is required. Fine-tuning a pre-trained model

is a common strategy to optimize a network on new classes. But also a large amount of labeled new input is required to avoid overfitting. Few-shot learning solves this problem by learning from a small dataset.

Few-shot learning methods have been mostly used in image classification tasks [SSZ17, SLCS19, KZS15]. Prototypical network [SSZ17] is designed for few-shot classification tasks. The network learns a metric space in which classification is performed on the smallest distance measure of the query sample to each of the prototype representations of each class. Huang [HWZ*19] performed robust time series classification for imbalanced data distribution by using a prototype embedding framework - Deep Prototypical Networks (DPN). They projected data to a main embedding space to capture the discrepancies of different time series classes. Due to the prototype representation of each class, they alleviated the issue of data scarcity. Evaluated on 49 time series classification benchmark datasets, they demonstrated the robustness of the proposed framework.

We consider this category to be the most realistic in designing applications for human activity recognition tasks with sensory data. Since we can not adopt to the complexity and diversity of all persons actions during the training phase, we need the network to have the ability to adapt to individual users by introducing only a small amount of this users data to optimize the trained network. Few-shot classification methods are methods that can be generalized to unseen classes, even when less labels from these classes are available without retraining the base model.

Exercise detection on personal devices is often applied to track daily ambulation activities such as *standing, walking, running*, or *stairs-up or down* [KWM11b, RDML05]. For tracking of more stationary activities involving whole-body interaction, a remote system is better than wearing a smartphone on the body, as the detection is more unobtrusive. However, the complexity and diversity in human action makes it difficult to develop one single system to fits all situations. To overcome this issue, it requires more advanced machine learning methods to improve the model generalization.

4.2.2. Database

The motivation and sensing principle of building such a database is already introduced in Section 3.1.7. Here we avoid repeating the details and only present the details of this dataset used to investigate the methods introduced in this section. This database enable us to study the body motion in relation to Doppler profiles from built-in hardware of different commercial smartphones. The effect of fine-grained movements from both limbs and arms cause micro-Doppler patterns in addition to the main Doppler reflection. Studying these micro-Doppler events enhances the ability of recognizing more complex and naturalistic human activities including whole-body interactions.

The presented database consist of two different setups, in order to investigate the effects of various methods on improving model generalization for different data distributions. Data of the first setup is called Lab-Data. The Lab-Data consists of data collected in our living laboratory as depicted in Figure 3.19 from 14 individuals. The group consists of 4 females and 10 males. Some general statistics about the test population are provided in Table 4.4. The built-in microphone of the sensing device is placed 50 cm apart from the exercising individuals on the floor aligned with the hip. For each individual, two separate sessions were collected, each has 10 repetitions of each exercise class. Left and right variations for exercises such as *segmented rotation, trunk rotation*, and *quadruped* are counted as one repetition. *Swim* is performed in average for 30 s in each session to reach similar time duration comparable to the other exercise types. In order to collect the micro-Doppler motions from the arms, the device is placed on the floor and aligned with the shoulder for the exercise of *swim* and *trunk rotation*. The duration of each session is approximately 7-9 minutes in average. The smartphone used for data acquisition has the brand Samsung Galaxy A6 (2018) and the placement of the sensing device to the exercising body is constrained to same position for all participants.

males	age	height	**females**	age	height
Number	10	-	Number	4	-
Min	21	172	Min	21	157
Max	33	193	Max	32	172
Average	25.4	182.3	Average	24.75	165.5

Table 4.4.: The statistics of the population of the participants for the Lab-Data.

Participant	Sex	Height	Exercise Frequency	Device	Location
P1	male	180 cm	Frequent	SONY Xperia XZ2 Compact	Environment 1
P2	male	181 cm	Frequent	Samsung Galaxy A5 (2017)	Environment 2
P3	male	181 cm	Frequent	SONY Xperia Z5 Dual	Environment 3
P4	male	182 cm	Frequent	Samsung Galaxy A6 (2018)	Environment 4
P5	female	168 cm	Less Frequent	SONY Xperia XZ2 Compact	Environment 1

Table 4.5.: Description of Software and Hardware Setups for the Uncontrolled-Data.

The goal of the second setup is to be leveraged on testing various finetuning approaches, as these data are collected under individual, different hardware, and uncontrolled environments independent of the Lab setup. This part of data is called the Uncontrolled-Data. It consists of data collected from five different individuals. Due to logistic and privacy constrains related to the experimental setup, the second setup contains a smaller number of participants compared to the Lab-Data. The hardware device is not limited to the Smartphone used in the Lab-Data. Each individual was asked to collect eight individual sessions distributed over several days in their familiar surroundings and without any supervision. Each session has a comparable length to the collected data from the Lab-Data. Some general statistics about the participants and the hardware devices used, are listed in Table 4.5.

The data acquisition app is installed on the individual mobile device. The participants from the uncontrolled setup were asked to collected data from their home environment to simulate the real-world application scenarios. Figure 4.11 illustrates the different data acquisition environments that affect the signal strength of the underlying hardware device. In contrast to the Lab setup, the other apartments all have wooden floors which makes the back reflected signal strength stronger compared to the Lab setup.

This work aims at investigating methods to improve the generalization ability of pre-trained models on new individuals under more realistic conditions. According to the underlying methods, we split the data as follows:

- Basic training data contain all individuals and sessions of the Lab-Data.

Environment 1 Environment 2 Environment 3 Environment 4

Figure 4.11.: Illustration depicts the four different data acquisition setups from the individual test participants.

- Subject development data contains 4 sessions (out of 8) of each of the users in the uncontrolled setup, the Uncontrolled-Data.
- Testing data contains the 4 sessions (out of 8) of each of the users in the uncontrolled setup, the Uncontrolled-Data, that were not used in the Subject development data.

4.2.3. Our proposed methods

Our application is built for use-cases in the domain of Quantified-self with mobile devices. This work is motivated by our observations from our previous experiences in realistic use of mobile device to detect workout exercises without using additional external hardware [FKK20b]. However, previous work faced a major usability challenge as it did not adapt well on individuals and environments unseen in the training phase. This issue is the main target of the methods presented in this section.

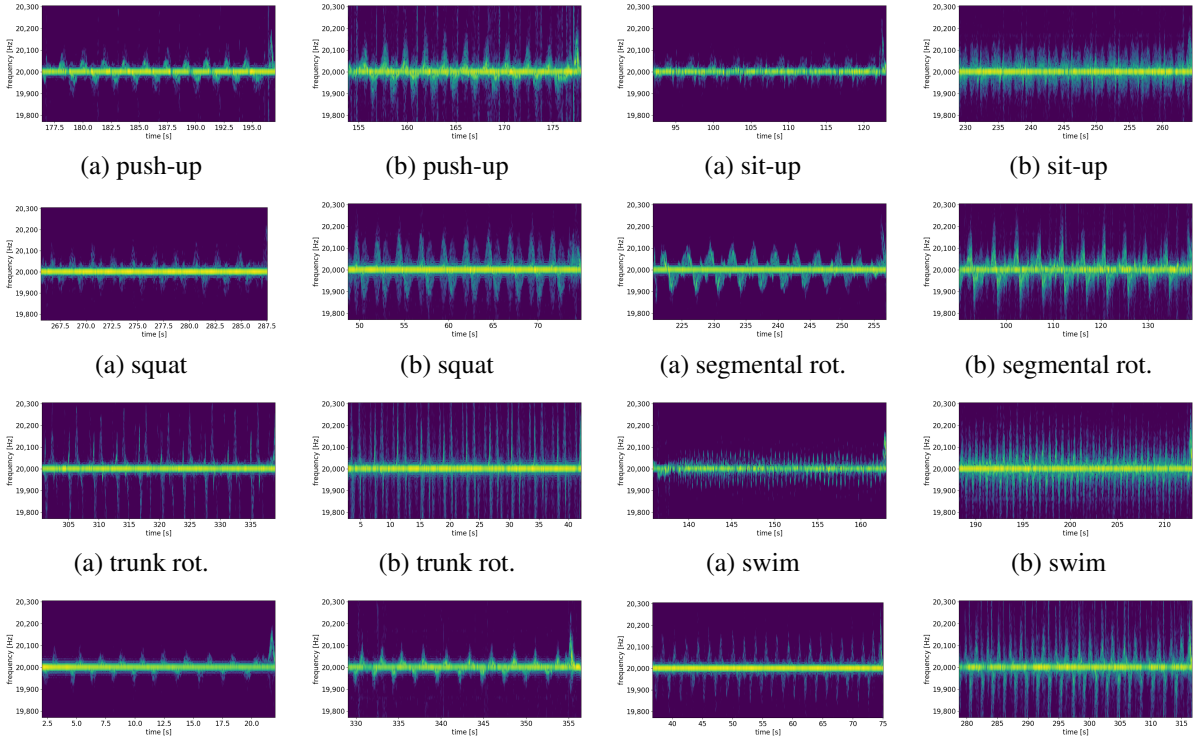
To state the problem, in Figure 4.12 (a) and Figure 4.12 (b), we illustrate the Doppler profiles from the same participant performing the same set of physical activities under two different environments with the same hardware and sensor position. Doppler profiles describe the patterns related to motion which can be extracted from the time-frequency spectrum. Figure 4.12 (a) is under the Lab setup and Figure 4.12 (b) is under the Environment 4 as depicted in Figure 4.11. The surface material where the sensing device is placed on has changed from carpet to wood. We observe that the signal differs for different environmental settings. Though the overall speed and form of the signals are quite similar, the strength and noise embedded into the signal reveal a strong difference in both settings, causing the performance of inference model trained under the Lab-Data to drop on the Uncontrolled-Data. This is mainly due to the material-dependent attenuation of the transmit signal.

The reason for the variability is due to the sensing condition. Due to different surface materials, the characteristics of the back reflection change. In case of the Lab setup, the sensing device is placed on a carpet causing a stronger attenuation, such that the back reflected echo signal is much weaker compared to place the device on a wooden floor.

In general cases, we can not train a classifier adapting to every possible sensing environment, unless our training data unrealistically contain unlimited possible variations. The quality of hardware devices integrated in the smartphone may also introduce strong variations in the transmitted and received signal power. But the physical sensing principle and the basic physical characteristics remain the same. To adapt to new, real-world circumstances, we need to individually finetune the trained model. In this section, we investigate several approaches to improve the model generalization on individual data.

4.2.3.1. The baseline method

The base inference model is built using a stacked bidirectional LSTM network as a baseline. To baseline our proposed solution, we need to demonstrate the exercises detection performance when the uncontrolled environment is not considered. The choice of using this sequence model is due to its ability to consider the global structure within a sample time window. The architecture of this sequence modeling network consists of 2 bidirectional LSTM layers with 128 hidden nodes in each LSTM cell. For each input node, a slice of the frequency bands (ranging from 19.5 kHz to 20.5 kHz) from a time step resolution (46.5 ms) is provided to the network. The bidirectional structure enables the network to look forward and backward in time to extract fine-grained sequence information from the spectrum domain. The overall time window of each sample is 6 s long. This parameter is set due to a preliminary study on the window length in regards to the classification performance. In Figure 4.13 (a), the bidirectional LSTM model is depicted with an two dimensional instance normalization layer applied



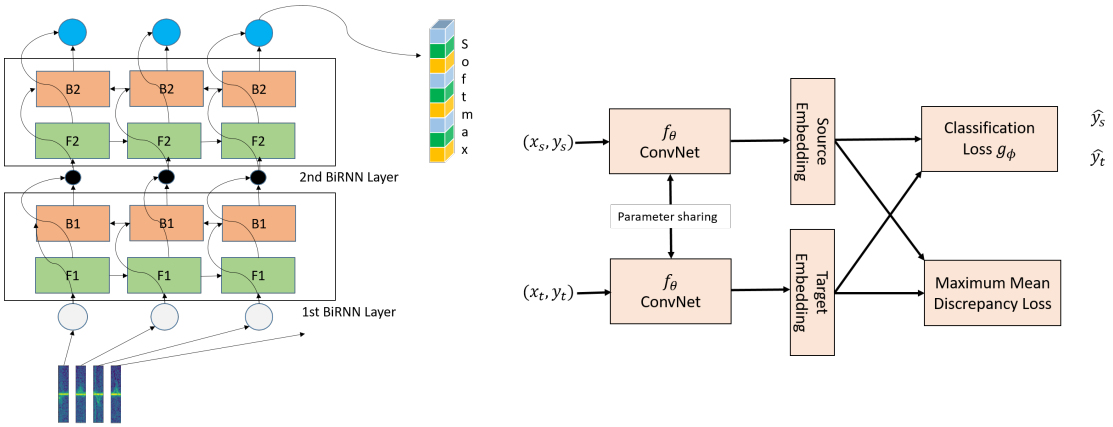
(a) push-up (b) push-up (a) sit-up (b) sit-up

(a) squat (b) squat (a) segmental rot. (b) segmental rot.

(a) trunk rot. (b) trunk rot. (a) swim (b) swim

(a) bridge (b) bridge (a) quadruped (b) quadruped

Figure 4.12.: (a) Visualization of the Doppler profiles collected in a constrained laboratory environment (Lab-data) for the Participant P4. (b) Visualization of the Doppler profiles collected in Environment 4 (see Figure 4.11) for the same Participant P4 from the Uncontrolled-Data.



(a) (b)

Figure 4.13.: (a) depicts the model architecture of the bidirectional LSTM. Each LSTM cell (B_i, F_i) contains 128 hidden nodes and two stacked layers are used to build the network. For each input node, a slice of the frequency bands (ranging from 19.5 kHz to 20.5 kHz) from a time step resolution (46.5 ms) is provided to the network. (b) depicts the domain adaptation model. Source and target data are projected into the same embedding space using the common base ConvNet. Adaptation is realized by minimizing the classification losses for source and target and reducing differences in both feature distributions.

prior to the network input. The number of output classes are nine that includes the eight true activity classes with the additional *none* class describing all the transitions and noisy samples between two successive action classes.

The learning rate is set to 0.001 and the Adam Optimizer is used to optimize the network parameters. Cross entropy loss is used as the cost function to minimize the loss of the misclassification error from the training samples. Batch-wise training is used, while in each batch, 15 samples of each class are randomly selected from the training data to construct similar training procedures compared to other network structures.

The training contains the Lab-Data only. Subject development data with 4 sessions from the Uncontrolled-Data is used in the validation stage, while the 4 sessions of the remaining Uncontrolled-Data is used in the test phase. Batch-wise approach is applied, while each batch contains 15 samples randomly selected from each class.

4.2.3.2. Our proposed method 1: Domain adaptation

Improving the model generalization ability, the first method we propose is from the domain adaptation (DA) network family. DA is commonly used to transfer knowledge from labeled source domain to target domain where data is unlabeled or only partially labeled. In our case, the source domain refers to the limited amount of data in the Lab-Data, while the target domain refers to the Uncontrolled-Data collected by the different individuals under changed conditions. The aim of such an adaption network is to make the distribution of the source and target embeddings in the common embedding space more similar, such that the classification network works well on both domains. By applying this architecture, we aim at adopting the base feature extractors to be sensitive to deterministic features from the Uncontrolled-Data domain. In such a way, the classification performance does improve for the uncontrolled setup.

The metric used to measure the similarity of both distributions is the maximum mean discrepancy (MMD) on the final feature level. The model architecture of the adaptation network is depicted in Figure 4.13 (b). The

usual approaches using domain adaption do not require to include target labels. However, without any label information from the target domain, the knowledge transfer does not work well on the targeted use-case, as both domains are quite dissimilar. In order to improve the knowledge transfer characteristic, we included partial target domain labels to increase the performance on feature level adaptation. By including partial labels, we include around 50 % of the labels from the subject development set of the Uncontrolled-Data. An instance normalization layer is used prior to the ConvNet to mitigate the hardware dependent effects of the transmit power from different smartphone models.

A base ConvNet is used to extract common features from source and target domain. The ConvNet structure consists of 4 successive convolutional layers, each followed by a batch normalization layer, leaky rectified linear unit (ReLU) activation layer with the negative slope coefficient of 0.2 and a max pooling layer to reduce the input dimensions. The successive layers increased the number of filters from 32-32-64-64. The same ConvNet structure is used for all networks as the feature extraction component in this work.

The embeddings in the embedding space are used to minimize the classification loss of the source and target domain. The measure to minimize the feature space difference is the maximum mean discrepancy (MMD) loss. Minimizing the MMD loss means to reduce the distance between the means of the two distributions and decrease the difference of both distributions. The objective is the combination of three different losses as given in Equation (4.11). The overall loss consists of both cross entropy losses from the source and target domain classification, and the maximum mean discrepancy loss of the feature embeddings originated from the source and target domain in the same embedding space.

$$\mathcal{L} = \mathcal{L}_{CE}(g_\phi(f_\theta(\mathbf{x}_s)), y_s) + \mathcal{L}_{CE}((g_\phi(f_\theta(\mathbf{x}_t)), y_t) + \mathcal{L}_{MMD}(f_\theta(\mathbf{x}_s)), f_\theta(\mathbf{x}_t))) \tag{4.11}$$

The network is trained with 100 epochs. Each batch composites of 15 samples drawn from each of the 9 classes. Adam optimizer with a learning rate of 0.001 is set to learn the hyperparameters of the network. Since the base feature extraction network needs to be fine-tuned on both data domains, a retraining of the base model is required.

We applied the Lab-Data as the source domain and the subject development set from the uncontrolled setup as the target domain. Both data are used in the training. The adapted model is evaluated on the Testing data of the uncontrolled setup.

This method is good to adopt the feature extraction layers to work for features from different domains. A negative aspect is that if both domains differ too much, it could lead to a negative adaptation and causing the performance on the source data to decrease. To address this issue, we present the following methods which do not need to retrain the pre-trained inference model to finetune to new individuals. We benefit from a few labels of the new datasets to label this unknown dataset.

4.2.3.3. Our proposed method 2: Few-shot classification

This section deals with three methods from the research domain of the few-shot classification learning. The network is trying to learn common features within a subset of tasks without retraining.

F1: Siamese Network with Few-Shot Classification The Siamese network consists of two identical feature extraction base networks with shared weight parameters. The learned feature embeddings from both inputs are then compared with each other to form a similarity score. Commonly it is used for verification tasks and the score indicates how similar two input samples are. Instead of the verification task as performed in common Siamese networks, we extended it for the few-shot classification task. In contrast to the domain adaptation method, this

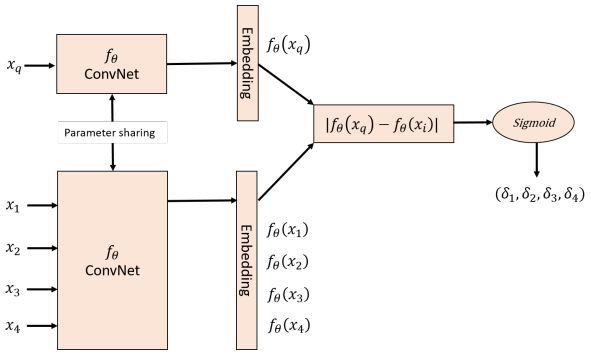


Figure 4.14.: The Siamese network for few-shot classification task is to learn the optimum separability for the multiclass classification problem. The ConvNet structure is used to generate feature embeddings. A distance measure is calculated for the query sample with all possible support classes. The output label is the highest similarity score within the multiple classes.

phase	epochs	ways	shots	query
train	500	9	5	15
valid	100	9	5	15
test	100	9	5-10	15

Table 4.6.: The parameter setup for the Siamese network is depicted here.

method can train on disjoint data. The Uncontrolled-Data is not used in the training stage. The working principle is depicted in Figure 4.14. Our network structure aims at learning the optimum separation between all multiple classes at once.

During training time, each training batch consists of 15 query samples from each class. The support set consists of 5 support samples from each class. They are fed through the feature extraction network ConvNet f_θ to get the embeddings. In case the support of each class is larger than one sample, a mean embedding is calculated to reduce the computational complexity. A similarity score is determined for the query sample embedding $f_\theta(\mathbf{x}_q)$ and the individual support class embeddings $f_\theta(\mathbf{x}_i)$, where $i = 1..C$ and C represents the number of classes. To determine the similarity score, an euclidean distance vector between the feature embeddings is calculated by $|f_\theta(\mathbf{x}_q) - f_\theta(\mathbf{x}_i)|$. This distance measure is further fed through a dense layer to learn the final similarity score $\delta_i = \sigma(\phi(|f_\theta(\mathbf{x}_q) - f_\theta(\mathbf{x}_i)|))$. This parametric dense layer is optimized by a pair of input samples every time step. Regarding the similarity measure between the query sample and all other support classes $(\delta_1, \delta_2, \delta_3, ..., \delta_C)$, the objective of the Siamese network is to maximize the maximum likelihood estimation or equally to minimize the negative log likelihood cost function.

During the training phase, each batch consists of 9 classes and 15 query samples each class. That means for $way_{num} = 9$ and $query_{num} = 15$ in total 135 samples is used for one batch training. To composite the support set for training, we randomly selected 5 support samples from each class, i.e. $shot_{num} = 5$. Adam optimizer is used to train the network parameter with a learning rate of 0.0005. In the evaluation phase, few-shot classification accuracy is leveraged. In Table 4.6, the setup for the training, validation and test stage is listed for overview.

The similarity measure of the Siamese network here is learnt by a parametric dense layer, in the next two paragraphs we introduce two other alternatives from the few-shot classification task where a non-parametric

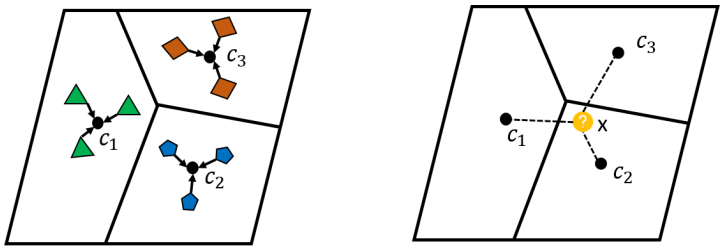


Figure 4.15.: Support samples from each class are projected to the same embedding space. The center of each support class is the average of the embedding vectors of the support samples in each class. The labeling for the new sample x is determined by the closest squared euclidean distance to the center of each support class. Comparing the distance of query sample to the mean embedding makes the processing faster. This image-based approach compares the locally connected features from the Doppler spectrum with each other.

distance metric-based learning is used in the classification stage. These methods are more robust against small difference in source and target domains, and thus more generalized.

F2: Prototypical Few-Shot Classification Network Inspired by [SSZ17], we adopted the prototypical network as the second method (F2) in the few-shot classification problem. In comparison to the first few-shot method (F1) with Siamese network, this network is non-parametric in the classification stage. This approach is a feature metric-based learning approach. Instead of transferring network parameters, this learning approach is based on learning the similarity distance between the feature embeddings from new target data to the prototypes of the samples from the same setup. This method is similar to a clustering-based method to find the k-nearest neighbours used for the classification.

The same ConvNet structure as in DA method is chosen to make the comparison across different classification models fairer while keeping the embedding extraction structure the same. We extended the structure of the prototypical network to adapt to our data and investigated the applicability of this concept on the task of individual finetuning with sensory data from our targeted application domains.

For each query sample and the samples from the support classes, they are projected by the same ConvNet network to the same embedding space. The average of the support samples from the same class is calculated to determine the center embedding of this class. The label of the unknown query sample is determined from the closest squared euclidean distance to the center embedding vector of the other classes. The described process is depicted in Figure 4.15. By combining the mean prototypical embedding, the processing and computation is faster. The mean embedding is valid under the assumption that the embeddings from the same class are situated close together in the embedding space.

The negative log-likelihood (NLL) function is applied to the negative euclidean distance of the embedding vectors to each class center embeddings. The objective is to reduce the NLL loss of the query output to the true classification label. Adam optimizer is used to optimize the weights of the ConvNet hyperparameters. A learning rate of 0.001 is used to learn the hyperparameters. In Table 4.7, the parameters for the few-shot learning and the batch-wise composition is listed for overview.

The disadvantage of using euclidean measure to express the similarity is that this measure is not bounded. In contrast to Siamese network, there is non parametric learning after the feature extraction ConvNet structure. The

phase	epochs	ways	shots	query
train	500	5	10	15
valid	100	9	10	15
test	100	9	5-10	15

Table 4.7.: The parameter setup for the Prototypical network for few-shot learning classification is depicted here.

classification is based on the k-nearest neighbour approach from the class embedding centers. The last method uses the cosine similarity measure which is a bounded metric.

F3: Local Descriptor Correlated Few-Shot Classification Network The third method of the few-shot classification (F3) is inspired by Li [LWX*19]. The author introduced a new distance metric to label the query sample. Instead of using image-level feature based measure, they introduced a local descriptor based image-to-class measure. We think this architecture will work on the 2D time-frequency spectrum, as this network is sensitive to local features. In contrast to object images, the Doppler spectrum contains less fine-grained contextual information, but local features caused by repetitive movements and micro-movements from limbs and arms are clearly observable. In this section, we examine the local descriptor correlation from the structural information extracted from the micro-Doppler range caused by different whole-body activities.

The local features from the ConvNet output is correlated with all other local descriptors of the support embeddings each class using a cosine similarity. This provides a locally feature-based image-to-class mapping. Advantageous of this approach similar to the prototypical network is that the classification network is also non-parametric. The learning is determined by the similarity measure of the most correlated local descriptors using cosine similarity. For all other support classes, the similarity score is calculated given $s = [s_0, s_1, s_2, ..., s_C]$. The label of the query sample belongs to the class with the highest support score. Instead of comparing query local descriptors to each of the support embeddings or the average embeddings individually, the top k most correlated local descriptors are extracted for each class from all support local descriptors within this class.

The advantage of the cosine metric is because the cosine distance measures the pattern similarity without being largely effected by the magnitude. The objective is to reduce the cross entropy loss for this multiclass classification problem. Adam optimizer with a learning rate of 0.001 is chosen. The parameters for the few-shot learning and the batch-wise composition is depicted in Table 4.7 for overview.

In few-shot classification learning task, the training data can be disjoint of the test data. We used the Lab-Data to build the training tasks and used the subject development data as support set and the testing data from the uncontrolled setup to evaluate the model.

In this section, we introduced three methods from few-shot learning: (F1) Siamese, (F2) ProtoNet, and (F3) LocalNet. These methods are theoretically suitable to enhance the generalization for our use-case, since we want to mitigate the retraining phase for similar tasks.

4.2.4. Evaluation and Discussion on our proposed Dataset

The database details are introduced in Section 4.2.2. In this section, we present and discuss the results of the proposed approaches in comparison to the baseline. We aim to optimize the trained model with Lab-Data under controlled condition to be generalizable to individuals under the Uncontrolled setups.

The results of the baseline model and models which require adaptation with the Uncontrolled-Data during the training phase are displayed in Table 4.8.

Method	Ratio of labels used from Uncontrolled-Data	P1	P2	P3	P4	P5
Baseline	-	16.25 %	46.35 %	11.17 %	25.73 %	15.5 %
DA	0 %	35,20 %	37,80 %	20.14 %	57.67 %	38.04 %
DA	50 %	77.61 %	85.39 %	58.61 %	78.86 %	75.87 %
DA	100 %	**87.13 %**	**98.14 %**	**84.10 %**	**76.92 %**	**96.85 %**

Table 4.8.: The accuracy results of baseline method and methods with model retraining is depicted. The term P_i indicates the ID of the participants. Allowing the knowledge from the Uncontrolled-Data to modify the base feature extraction increases the performance on individual finetuning.

4.2.4.1. Baseline results

Given the Lab-Data in the base training, the trained inference model does not generalize well on test data collected under uncontrolled conditions, if not provided in the base training. The stacked bidirectional LSTM network failed to cope with data from real-world environments, as both data distributions differs too much. According to the baseline result provided in Table 4.8, in most of the cases the accuracy equals to a uniform distribution. The model performs no better than random guessing on the Uncontrolled-Data. In our application, the total number of classes is nine, random guessing corresponds to an average accuracy of $\frac{1}{9} = 11.1\,\%$.

4.2.4.2. Our proposed approaches

The results of using the domain adaptation (DA) method are depicted in Table 4.8. Here we successively increased the amount of target labels from the subject development set to be included into the training to improve the performance of the domain adaptation. Without including any target labels, the network only minimizes the distribution of the source and target embeddings in an unsupervised way. The performance is only 10-20 percentage points better compared to the baseline. By incorporating more label information from the target domain, the knowledge transfer improves significantly on the target domain classification. With only 50 % of the target labels, the results increased about 40 percentage points and with 100 % of the labels, the results increased about two to six folds in average.

Results of Few-shot Classification Networks Here, we evaluated three alternatives of the few-shot classification approach. These models typically do not need to retrain the pre-trained inference model, as the training data are disjoint. An overview of the evaluation results is given in Table 4.9. A general tendency is that the classification accuracy increases at least 5 percentage points with increasing number of support samples per class used in the evaluation. This is trivial, due to an increased reliability and an improved decision boundary related to more support samples. The Siamese network (F1) provides similar results compared to the prototypical network (F2), as both network architectures work with euclidean similarity scores on the sample-base. Their performances are increased around 50 percentage points compared to the baseline model. The image-to-class measure in the LocalNet (F3) performs the best as depicted in Table 4.9 with an increase of two to six folds in average compared to the baseline.

The training of few-shot classification can be performed on disjoint data. Therefore, no knowledge of the uncontrolled setup is to be included in the training phase. This way, no retraining is required. The method is similar to k-nearest neighbour classification with the appropriate distance measure using support samples or correlated sample embeddings.

No.	Method	P1	P2	P3	P4	P5
-	Baseline	16.25 %	46.35 %	11.17 %	25.73 %	15.5 %
F1	Siamese@5	65.53 %	83.77 %	72.81 %	65.61 %	68.33 %
F2	ProtoNet@5	67.06 %	79.9 %	67.85 %	77.54 %	65.92 %
F3	LocalNet@5	**85.28 %**	**79.9 %**	**64.85 %**	**94.33 %**	**86.98 %**
F1	Siamese@10	69.69 %	90.07 %	75.65 %	69.08 %	72.56 %
F2	ProtoNet@10	74.18 %	88.59 %	69.31 %	87.88 %	70.80 %
F3	LocalNet@10	**89.55 %**	**97.84 %**	**67.51 %**	**98.00 %**	**91.63 %**

Table 4.9.: The accuracy for the three alternatives of few-shot classification task is shown. The @number indicates how many support samples each class were used in the evaluation. The term P_i indicates the ID of the participants. The performance increases in general with the increasing number of supports. The LocalNet with the cosine similarity measure outperforms the other methods, as it includes the local to global feature correlation.

4.2.4.3. Discussion and some remarks

The performance of activity recognition based on ultrasound sensing using a mobile device is subject to many variables. The same hardware applied under changed conditions for the same person show variability in the signal strength. To deploy a fixed application to real-world scenarios is therefore not easy and often has to overcome some difficulties. In many cases, the performance drops due to the dissimilarity in both domains. To overcome this issue, we investigated several methods in this section. We provide a database collected under various conditions to allow researchers perform experiments on it to solve the problem of lack of generalization on new individuals. The base data consists of Lab-Data under controlled environment and same sensing device. Uncontrolled setups from five different individuals are used to evaluate the methods for finetuning on individual dataset.

Finetuning a base model on new domains requires sufficient amount of labeled samples from the Uncontrolled-Data. Otherwise, the finetuned network would overfit adopting on this small data amount. However, labels are most difficult to acquire and the individual labeling process might be error prone. In such cases, domain adaption method can be leveraged, where no label information of the target domain is required. Though, such network could benefit from including a small amount of the target labels in case both domains differ as investigated in our use-cases.

Few-shot classification is suitable for adopting finetuning on limited data without retraining. This method can cope with individual hardware characteristics without modifying the base training. By comparing knowledge extracted from support samples of different categories, an unknown sample is able to assign to the correct category under the assumption that samples of similar categories are close in the embedding space. As no feature adaptation from the target domain is applied, this model requires both domains behave similarly.

To leverage few-shot classification, the user has to pre-label a small amount of individual sessions before the model is adopted to this user. These labels are used to classify the new samples based on certain distance metrics. The developer does not need to modify the feature extraction network to individually adapt to each new user. In case of the domain adaptation, the developer needs to modify the base feature extraction according to the user data. It further assumes the similarity of both domains in order to avoid negative knowledge transfer.

4.3. Summary

In this chapter, I address the research question 5 *How to overcome the gap between constrained development data and the more complex **real-world data** with the scope on time series for HAR applications* by exploring various methods to improve the model generalization ability and robustness to cope with real-world scenarios. Bridging the gap between development data and real-world data, it often requires the model to be finetuned, as it is improbable to create a generalized model to fit all circumstances.

Regarding the evaluation of the methods proposed in this chapter, I first presented two different databases using two applications developed in the preceding chapter. The first database consists of time series collected from the capacitive proximity sensor prototype developed in Section 3.2.3. These multivariate time series acquired from the coordinated whole-body exercises are used to investigate the data augmentation techniques on the data space. Traditional data augmentation such as flipping or rearranging time series can be applied given the benefit of the symmetrical system setup. The second database consists of Doppler data using built-in hardware of commercial smartphones (in Section 3.1.7) under a controlled and various uncontrolled setups. This dataset is used to investigate methods such as domain adaptation and few-shot classification to improve the model generalizability for different users under uncontrolled setups. These methods work with metric-based approaches on the data embedding space and are especially well adapted for limited data amount as in the case of most human activity recognition tasks.

With respect to the gathered knowledge from these experiments, I can draw the following conclusions: In case of the smooth multivariate or univariate time series, such as capacitive proximity sensor data, data augmentation technique (in Section 4.1) can be directly applied on the data space. Conventional methods, such as magnitude and time warping introduced diversity in time series data and can decrease the model bias on test data. Generative approach can be leveraged to synthesize samples to further improve model generalization ability and reduce model variance due to introducing small variations in the input data. However, data augmentation requires development signal and real-world signals be drawn from the same distributions. Generative models create samples approximating this underlying data distribution.

In case the condition of the same distribution is not given, other methods manipulating the data embedding spaces have to be leveraged. These methods are such as domain adaptation or few-shot learning. The inherent difference between development data and the real-world data is the more realistic use-case. Domain adaptation method (in Section 4.2.3.2) can overcome the domain difference by adapting the base feature extraction domain to minimize the embedding distributions from both domains. In this way, the source classifier works well also for the new target domain. However, the domain adaptation method needs retraining to adapt the base feature extraction layer.

In order to avoid retraining, few-shot classification (in Section 4.2.3.3) can be leveraged. The base training is on sub-tasks for the feature extraction network to learn the types of features they are expecting to see. The classification is adopted on similar new tasks without having been trained on these tasks before. By only using a few samples from the target data, the model is able to classify unknown samples from the same target domain with respect to an appropriate distance metrics. Few-shot learning is especially suitable for limited data amount. This approach protected the data privacy and facilitates the fine-tuning process on new individuals.

There exist other advanced machine learning methods which can be applied for improving the model generalization ability on uncontrolled real-world scenarios. They are far too large to cover all and therefore difficult to make a reliable recommendation. Of necessity, here is just a collection of methods, I have explored to solve special issues regarding the underlying sensor applications and time series properties.

5. Online application

In previous chapters, we have dealt with the problem of sensor selection, data acquisition, data processing and data modeling to build successful sensor-driven applications for activity recognition. However, until now these steps are mainly performed on offline data. Offline processing enables us to develop algorithm according to the recorded data. The ability of investigating various methods or models is only possible using offline processing. In this chapter, I answer the RQ6 *how to scale model complexity used in real-time applications* by targeting the problem of deploying online application, assuming the offline processing resulted in a successful model.

After the model has been trained and finetuned for specific task, the last step is to deploy it to a production system. The production system can be either a server in the cloud, on a working desktop PC, or on the processing device itself in case of edge computing. For edge computing, the computation resources and storage are often limited due to the restriction of hardware capacity. Therefore, in this chapter my research contributions to this research direction is categorized as follows:

- Contribution 1: Deploy exercise recognition with capacitive proximity sensors on a Raspberry Pi 3 (Section 5.1)
- Contribution 2: Design algorithm/model with capacitive proximity sensors for mid-air gesture recognition on standalone device with limited processing capability (Section 5.2)

In Section 5.1, I show the use-case of deploying the offline developed model in Chapter 3.2.3 for exercise recognition and counting to a Raspberry Pi 3 with the main focus on online processing. The design choices, setup and the offline model development are referred to Chapter 3.2.3. In this section, I deploy the trained model direct on the Raspberry Pi 3 and present the components used for real-time application.

In Section 5.2, I propose a mid-air gesture recognition using capacitive proximity sensors. I introduce the design of a simplified model-based algorithm to use it on a standalone device with limited computation resources. The hardware prototype used is called Rainbowfish [GPBW14], which is a capacitive touch screen device with sparse spatial resolution. The system allows proximity sensing with 12 capacitive loading mode sensors.

5.1. Deploying an exercise recognition model on a Raspberry Pi 3

The application referenced here is proposed in Chapter 3.2.3. It is an enhanced yoga mat with eight capacitive proximity sensors to measure whole-body strength-based exercises. In this section, we propose the extension of the proposed application to be deployed to an online real-time application on a Raspberry Pi 3.

After the offline processing for developing a well trained model derived from the collected dataset, the final goal is to develop a system that is suitable for a real-time application on low resource devices. To demonstrate this aspect, a simple web server application was implemented on a Raspberry Pi 3. Since modern smartphones have more computing capability the application could be ported to these platforms without any performance impairments.

The application was only implemented to use the CNN models, as it was empirically clear that the evaluation time is much faster. Feature extraction in the offline fashion was implemented by using the library toolkit *tsfresh*.

It was not intended for real-time applications but rather for offline analysis [CBNKL18] which implies that the feature extraction would have to be re-implemented more efficiently. Examining the time consumption of each component along the processing pipeline, we noticed that the handcrafted feature extraction was the most time consuming one. Avoiding this step would reduce the processing time and makes the application more real-time capable. A thorough evaluation in previous sections for the user-dependent and the user-independent use-cases reveals the strength of using CNN model towards conventional machine learning models. The CNN model is an end-to-end training method, which connects the feature extraction and the classification in one optimization step without the need of building handcrafted features.

In addition, one of the major benefits of artificial neural networks is the property, that the evaluation time solely depends on the complexity of the architecture, i.e. the number of nodes and connections. Once the model architecture is finalized, the computation remains constant for increasing data size [Yos12]. Also, the prediction of unknown data points is very efficient, as a feed-forward operation only consists of matrix multiplications which are heavily optimized in the used libraries [C*15] and can be performed in batches to further improve the efficiency. While optimizations for conventional classifiers exist, e.g. for k-NN the evaluation time grows when the data pool increases (which also decreases the suitability of introducing synthetic data). Support vector machines (SVM) also dependent on the number of supports which scales with the underlying data pool. However, relate to traditional machine learning approaches, the greatest bottleneck remains the models dependence on the aforementioned feature extraction and data amount.

5.1.1. Components

The basic application concept is depicted in Figure 5.1. The opencapsense (OCS) board is used to publish the sensor readings via a Bluetooth chip operates at a baud rate of 115200 Bit/s. It is powered by a power bank and paired with a server system that reads the data for further processing. During the data acquisition and the development phase, the paired system was a laptop. This allows us to store data without having the storage limitation issue. There was no difference observed between using Bluetooth or USB for data collection, due to the moderate transmission rate and as these system components were always placed in the same room and positioned only 2-3 meters apart.

The server has access to the trained CNN model and calculated templates from the training data. The model is used to classify the current activity, while the templates are used to build correlations with the current signal to count the repetitions of the performed exercise. The server hosts a simple website which is used to display the current activity and repetitions. The website can be accessed by the clients via the internet, such as a desktop PC, smartphone, or a laptop PC.

To simplify development, a simple input simulation interface was implemented. The train data saved in text files is read and corresponding to the recorded time stamps passed in intervals to the the input data array. As a consequence the OCS board does not have to be connected to the server to develop the application and we can *replay* recorded sessions. In real-time application, the sensor values are collected via the serial communication interface of the OCS board.

5.1.2. Implementation

The server side code is implemented in Python using the *flask* package, which allows us to remain in the same ecosystem and prevents rewriting necessary parts of the processing pipeline for a new environment. By powering up the Raspberry Pi 3, it loads the CNN model and tries to connect to the OCS board. A simple webpage

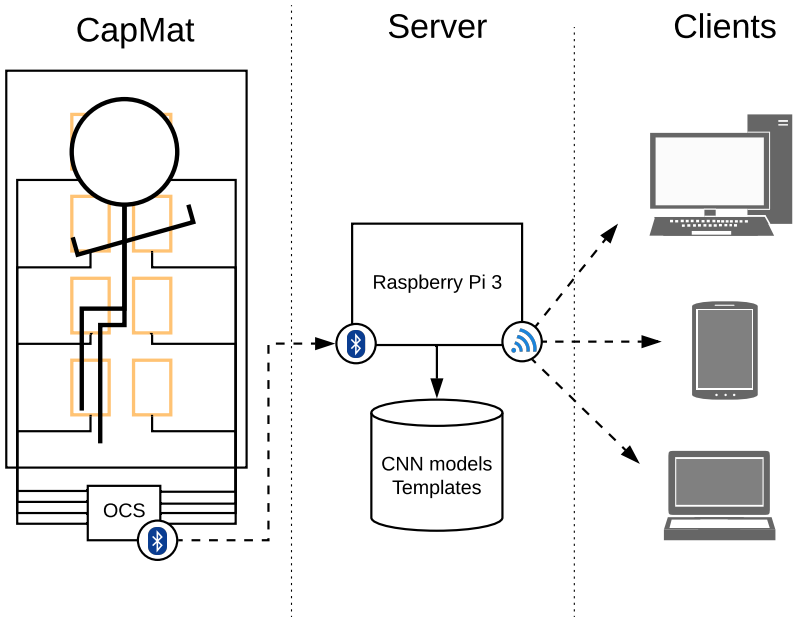


Figure 5.1.: The setup of a simple client-server architecture using Bluetooth and W-LAN as communication mediums. Figure refers to [LJ18].

displaying the current activity and the corresponding repetition count is hosted. The webpage is accessible for all clients that are connected to the same WLAN and know the IP address of the server.

The *apscheduler* package is used to run multiple jobs in parallel. The multi-threading procedure is necessary, as we want to continuously read incoming data from the OCS board into a buffer and to process and classify new data in parallel. In order to save processing resources on the server side with only limited processing capability, we only process new data every second. As such, the task is split into two separate tasks. First, the server connects to the OCS board and starts to read available data. Second, it extracts the last 6 s, corresponding to our determined window size, of stored values to process and classify them. The window length of 6 s is determined with respect to optimize the performance on the acquired development data. The first job is continuously checking for new data from the OCS board, while the second job is triggered every second only.

The processing job entails resampling the data after slicing the last 6 seconds from the buffered data readings to reduce the issue of missing values, so that we obtain the required 120×8 window format. No additional normalization step is required as further processing step, as the instance normalization layer is incorporated in to the first layer of the feature extraction network. If the newly predicted class is not *None* but any exercise, the data readings are stored in another buffer. This buffer is used to correlated with the exercise templates for the counting task in the next step.

For template matching, we can select the respective template from the classification result and move it over the stored data readings that belong to the exercise. This processing step is called the normalized cross-correlation. With the empirically determined thresholds in our previous experiments, we can simply count peaks where the similarity with the template maximizes. Since we need to perform matching four times in all orientations, this step proved to be computationally expensive, especially on the Raspberry Pi 3. Calculating the matches for

the whole buffer every second was not feasible in real-time, but to some extent rematching and counting peaks multiple times is necessary to ensure that we don't count repetitions twice. The sliding window approach could result in multiple peaks from the same exercise. If the newly arrived window segment just ends while a repetition is being executed, it will result in a peak – but the next window segment will also contain a peak, as it captured the continuing part of the same repetition. This should be taken into account. To further reduce the heavy computation of template matching, we heuristically determined to rematch the last 60 samples each time, i.e. 3 s of data, instead of just 1 s.

The communication of the currently active predicted exercise and its corresponding repetitions is established by using the WebSocket protocol. Whenever a new client opens the hosted web page, a web socket is opened. This mechanism allows the server to broadcast messages to all connected clients. This seemed more suitable, as the server only needs to broadcast messages when there actually are changes in the recognized activities. The alternative would be that the clients continuously fire requests to receive updates, which would increase the communication load on both client and server. While the application only displays the current activity and its repetitions, it could be extended to permanently store previous sessions.

In Figure 5.2 a quasi-live classification results subject to replay sessions of some randomly selected test participants from our preliminary test study is illustrated to demonstrate the working principle of our proposed application.

To summarize, this application benefits from mitigating the handcrafted feature extraction step to improve processing time. By choosing feed forward network, the model size is fixed in comparison to traditional machine learning methods which scale with the amount of training data. In the next section, we propose a small model-driven application to be implemented on a standalone device with restricted hardware capacity.

5.2. Running mid-air hand gesture recognition on a standalone device

This section aims at developing computationally inexpensive algorithms that can be implemented on low-cost embedded systems for edge computing purpose. Here we implemented a model-based approach by leveraging the dynamic time warping method (DTW) combined with bag of words (BoW) approach for mid-air gesture recognition with capacitive proximity sensors. DTW is a common approach used to measure the similarity of two sequences and thus ideal to work with sequences of time series. However, a time series can be of different length depending on the sampling frequency of the underlying system and the observation time window. If applying dynamic time warping directly on time series, the computation could be time consuming. Therefore my idea is to convert a lengthy time series to symbolic strings which are compared to the bag of words in a code book for the predefined gestures.

The following section contain extracts from the work published in [FGPK15]. The underlying capacitive sensing system, for which the sparse implementation of gesture recognition is designed, is called Rainbowfish [GPBW14]. The physical sensing principle is the capacitive loading mode. A prototype is depicted in Figure 5.3. It consists of 12 transparent electrodes each serving as a capacitive proximity sensor. The overall proximity sensing surface of the Rainbowfish has a dimension of 40 cm $\times$ 25 cm containing 12 rectangular transparent electrodes used for determining the position of a human hand. The system can also provide instant visible feedback according to the user actions performed by leveraging the LED lights integrated beneath the transparent platform. Object localization above the sensing surface is performed by using a straightforward weighted averaging method, which offers a fast way of position estimation. The estimation of the 2D position is given by Equation (5.1),

$$\mathbf{x} = \frac{\sum v_i \cdot \mathbf{x}_i}{\sum v_i} \tag{5.1}$$

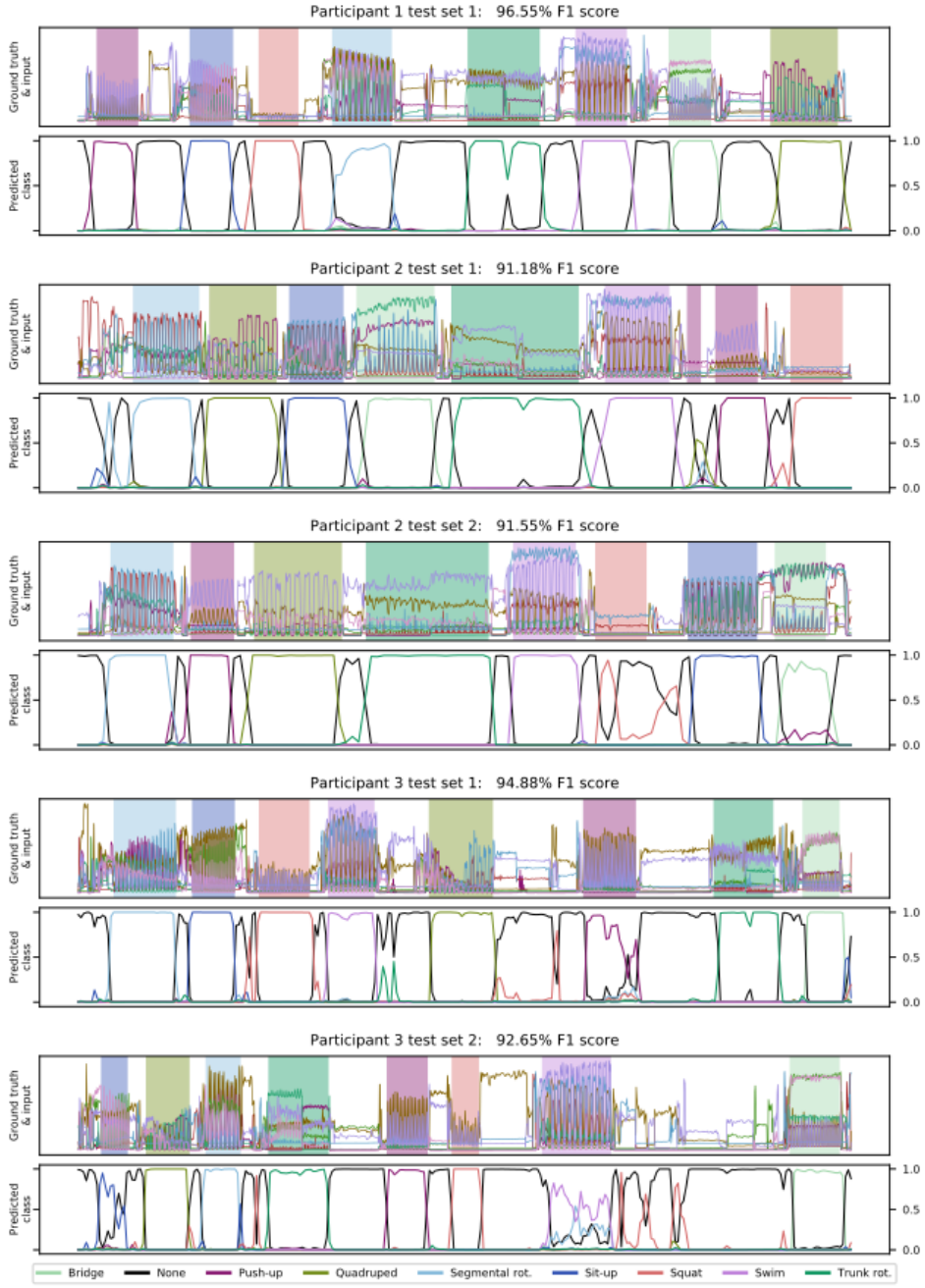


Figure 5.2.: A quasi-live classification results subject to reply sessions is illustrated over all predicted class probabilities by the CNN model on unseen test dataset over time. The color corresponds to the true labels marked as rectangle patches of the same color. Illustration originates from [LJ18].

Figure 5.3.: The prototype of Rainbowfish is depicted here as an application combined with door opener functionality. Users are able to open and close the door using swipe gestures.

Symbolic	Gestures
⇇	Swipe left to right
⇉	Swipe right to left
↑↑	Swipe upwards
↓↓	Swipe downwards
↻	Circular Movement clockwise
↺	Circular Movement counter clockwise

Table 5.1.: A list of all recognizable single handed gestures using dynamic time warping method.

where v_i stands for the capacitance value of the measuring electrode, $\mathbf{x}_i = [x_i, y_i]$ are the center position of the capacitive sensor plate and $\mathbf{x} = [x, y]$ is the 2D hand position in the air. In order to provide a smoother localization, a moving average filter on the 2D position estimation is leveraged.

The next major step is the gesture recognition and its interpretation, in order to make interaction between user and their environment possible. With our proposed method, we can quickly and almost confidently detect a set of simple hand gestures by leveraging the traditional dynamic time warping method. All recognizable gestures are intended to target single handed gestures. A list of the available gestures are given in Table 5.1.

5.2.1. Dynamic time warping with univariate time series

Now we outline the detailed processing steps for the gesture recognition task. The method of dynamic time warping is used to compare two symbolic time series, while one of them is extracted from a template database of reference hand gestures. In order to find the best match of a given time series compared to a template database, a cost function is calculated. The score indicates the similarity of these two time sequences. The best match with the highest score, or the lowest cost, is the predicted hand gesture from the predefined database. The mapping is performed in a nonlinear fashion, since the length of a performed gesture can be varied which depends on the

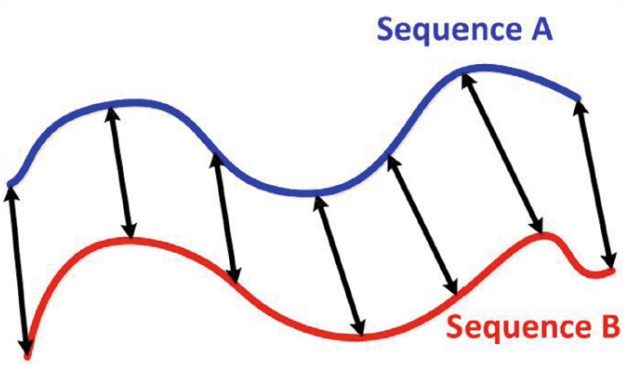


Figure 5.4.: Figure sketches the method of dynamic time warping on time sequence data. Two different sequences A and B are aligned in an optimum path using minimum score.

gesture's speed. Therefore, the two time series should be non-linearly scaled in order to optimally match each other.

Following this approach brings in one constraint: the first and last element of both time series should be mapped together. This is called the boundary condition. Suppose we have two time series A $= (a_i)$ with the index $i = 1..N$ and B $= (b_j)$ with the index $j = 1..M$, where the length of both sequences could differ. We are looking for an optimal path between these two sequences with the smallest score, whereat (a_1, b_1) and (a_N, b_M) should be mapped together. The concept is illustrated for two continuous time series in Figure 5.4 for clarity. The score matrix is of the dimension $N \times M$ and is built by comparing element of both time series with each other. All possible paths through the score matrix will results in viable solutions. The sum with the smallest score difference is the optimum solution. Different measures can be used to determine the element in the score matrix. One possible solution is to use the euclidean distance. Error measures or other self-defined metric can be adapted to the individual needs.

5.2.2. Implementation

For this sensing platform, a time series of hand positions is sampled into a discrete sequence. Depending on the duration above the sensing area, the gesture can be of different lengths. However, the computation is too expensive, if DTW is directly applied to the sequence of gesture locations, since these locations are discrete over the entire sensing area and not nominal. Therefore, we convert each sequence within the gesture trajectory into symbolic features, which is used to conduct the time warping method with a code book of stored simple gestures. The feature representation is illustrated in Figure 5.2 with a circular chart diagram. The radial component of this circular chart represents the velocity component of certain part of the individual gesture. One single gesture is sampled in subsequent hand positions above the sensing area. From one sample point to the successive sample point the velocity component is calculated. If it is below a certain threshold, it will be interpreted as an indecisive slow movement and will be represented with the character **Z**. Otherwise, the angular movement of the velocity component will be calculated and mapped adequately to the appropriate angular symbolic character. The start of the gesture is set, if the user's hand is above the sensing area and thus the starting command will be filled with a character **S** symbolizing the start of this gesture. The ongoing gesture is evaluated as long as the gesture

Character	Definition
S	Start of a gesture
Z	Slow velocity component between two successive parts of a gesture
D	Angular component indicating horizontal movement from left to right
B	Angular component indicating horizontal movement from right to left
C	Angular component indicating vertical movement up-down
A	Angular component indicating vertical movement down-up
E	End of a gesture

Table 5.2.: The meaning of the characters used to convert positional gesture information into nominal information used for dynamic time warping method.

can be recognized and the final termination of the determined gesture can be set by leaving the sensing area. As soon as the user's hand leaves the sensing area, the end character **E** will be added to the command stream. The termination character **E** can also be generated when the hand remains above a certain point for a longer time. This ensures that there is no obligation of the hand leaving the sensing surface. The definition of the used symbolic characters can be found in Table 5.2.

The graphical interpretation of the angular distribution with respect to their corresponding symbolic string characters can be seen in Figure 5.5. Due to the geometric property of the sensing area, where the length is broader than the width, it is reasonable to chose the angular distribution such that it is in favor of the horizontal movement. Favored by the larger x-axis with respect to the y-axis, the user has more freedom and precision by performing horizontal swipes.

The software realization can be found in the flow chart in Figure 5.6. The capacitive sensors keep actively measuring the activities above the sensing area. Once it detects the presence of a user's hand, the start character **S** will be added to the command sequence. Afterwards it keeps reading sensor values to update the gesture. The corresponding string command keeps appending to the existing command sequence. The algorithm keeps detecting the gesture performed by the user in real-time, as long as the user's hand does not leave the sensing area. As soon as the user's hand leaves the sensing area, the last gesture will be analyzed and the command sequence will be cleared, such that the system will be ready for a new gesture.

An exemplary template for horizontal gesture moving from left to right can be expressed by a sequence such as *SDDDE*, whereas the noisy real-world sequence may look like *SDDDZZDDDE*. Thus, the temporal stretched real-world sequence will be compared with all possible reference command sequences. The reference gesture with the lowest score and thus the highest matching score is the predicted user gesture. One special cost function and its distance function is given in Table 5.3 and Table 5.4. The penalty measure used here is the comparison score. A penalty of 1 is added if both actions represented by the symbolic characters differ, otherwise 0 is applied to the score matrix. The distance matrix given in Table 5.4 lists all possible accumulated penalty for the two sequences. The path with minimum accumulated penalty is the preferred path and gives the final matching score of both symbolic sequences. As observable from the example above, the only penalty is introduced from the noisy input **Z**. To decrease the penalty, we could ignore the penalty introduced by such noisy inputs.

The code book with the predefined gestures depicted in Table 5.1 is used to compare to real word sequences. The gesture with the lowest distance, equivalently the highest matching score is selected.

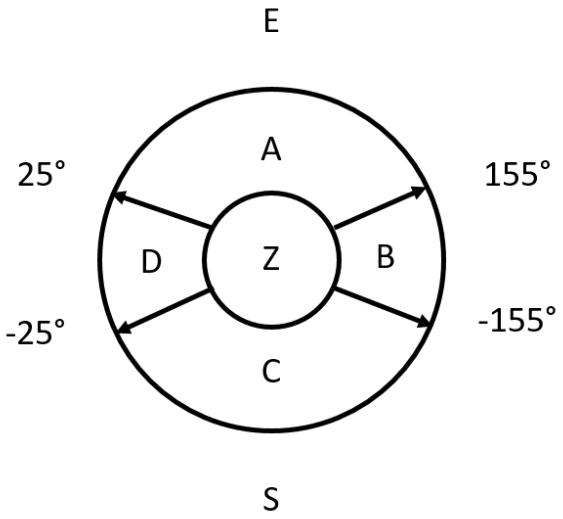


Figure 5.5.: The figure illustrates the way how the tangent of relative movement is mapped to the respective string character. A character **S** will be added, when the user's hand is detected on the sensing area for the first time and the character **E** will be added when no object is above the sensing plane. As long as the relative movement is small, the character **Z** is added, otherwise the other characters in the circle chart will be appended accordingly.

E	1	1	1	1	1	1	1	1	1	0
D	1	0	0	0	1	1	0	0	0	1
D	1	0	0	0	1	1	0	0	0	1
D	1	0	0	0	1	1	0	0	0	1
D	1	0	0	0	1	1	0	0	0	1
S	0	1	1	1	1	1	1	1	1	1
	S	D	D	D	Z	Z	D	D	D	E

Table 5.3.: The cost function showing the penalty element between both sequences. The vertical sequence is the template from the code book, while the horizontal sequence is the real world gesture sample.

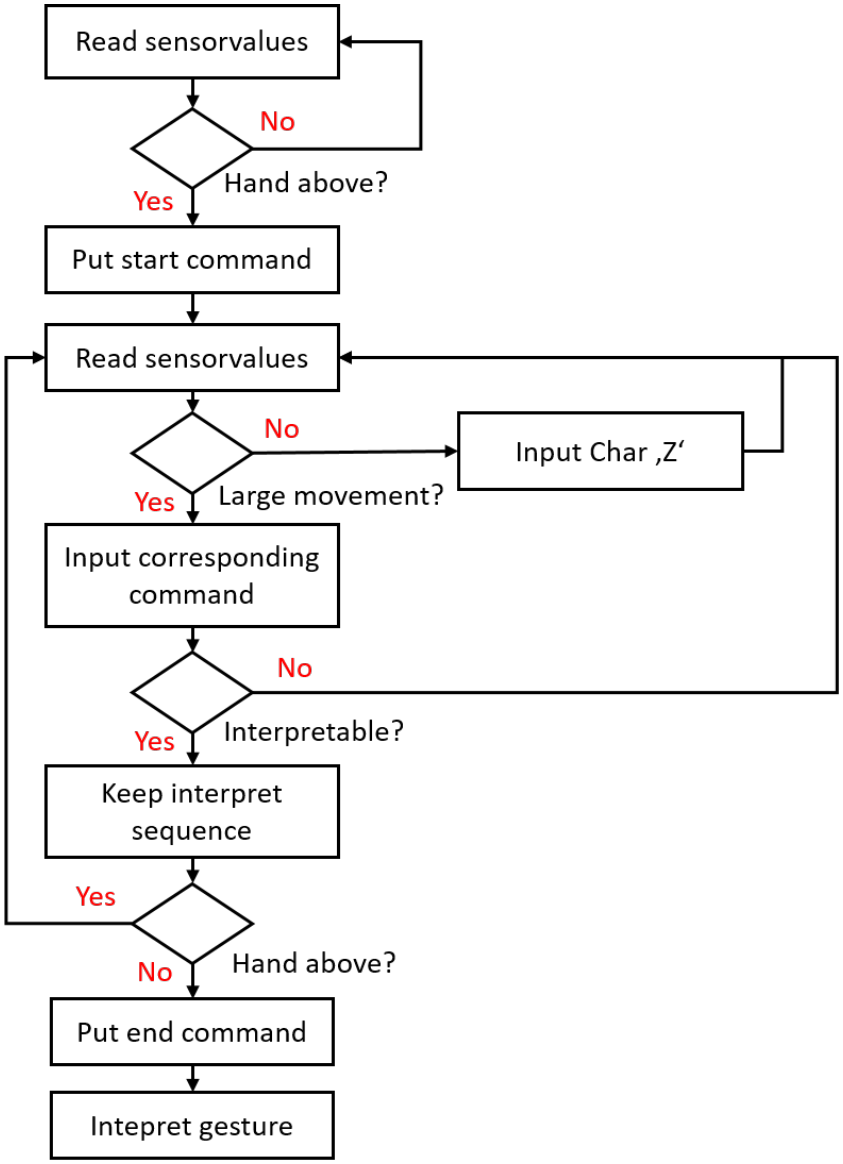


Figure 5.6.: The pipeline of the software implementation is depicted in the flow chart.

E	inf	5	1	1	1	1	2	3	3	3	**2**
D	inf	4	0	0	0	1	2	2	2	**2**	3
D	inf	3	0	0	0	1	2	2	**2**	2	3
D	inf	2	0	0	0	1	2	**2**	2	2	3
D	inf	1	**0**	**0**	**0**	**1**	**2**	2	2	2	3
S	inf	**0**	1	2	3	4	5	6	7	8	9
	0	inf	inf	inf	inf	inf	inf	inf	inf	inf	inf
		S	D	D	D	Z	Z	D	D	D	E

Table 5.4.: The distance function is depicted here. The vertical sequence is the template from the code book, while the horizontal sequence is the real world gesture sample. The matrix elements are the accumulated penalty comparing both sequences. The path with minimum accumulated penalty is the preferred path and gives the similarity score. Here the best path is marked in yellow.

	⇉	⇇	↑↑	↓↓	↻	↺
⇉	92	0	5	6	0	0
⇇	0	96	0	0	1	1
↑↑	4	3	95	3	0	0
↓↓	4	1	0	90	0	0
↻	0	0	0	1	98	1
↺	0	0	0	0	1	98

Table 5.5.: The confusion matrix for evaluation is shown. The diagonal elements indicate the true positive samples, while the off diagonal elements indicate the false recognition.

5.2.3. Validation and interpretation

Based on a user study conducted with 10 different test persons, we evaluated the feasibility of our proposed method. Each test person was supposed to execute the presented gestures given in Table 5.1. Each gesture was performed ten times above the sensing area. The result is evaluated and summarized in the confusion matrix, which is shown in Table 5.5.

From the confusion matrix given in Table 5.5, we observe that the circular movements are detected with a true positive rate of more than 98 %, while the other simple linear gestures achieved a true positive rate varying from 90 % to 96 %. It is quite apparent, that the performed circular movements clockwise or anticlockwise are recognized with very high accuracy, while the simple linear movements are less accurate. Simple linear movements are more error prone, since the capacitive sensing area is too sensitive, that it measures every tiny movements above the sensing area. The recognition rate increases, if the gestures are clearly performed. However, this problem of noisy input signals can be resolved, if we reduce this penalty of indecisive movement to 0 instead of 1. The precision and recall matrix is given in Table 5.6 to show the overall performance measures of the system. A precision and recall rate of more than 98 % is observed for actions with circular movements, while for simple linear and directional movements a precision and recall rate of 90 % are achieved.

In this section, the proposed mid-air gesture recognition was successfully realized by using dynamic time warping method. Dynamic time warping method on univariate symbolic time series with the bag of words approach is fast. A preliminary user study is conveyed and the results are evaluated and summarized. It showed that the circular movements clockwise or anticlockwise can be detected with very high accuracy, while the

Gestures	Precision	Recall
Swipe Right	0.89	0.92
Swipe Left	0.98	0.96
Swipe Upward	0.91	0.95
Swipe Down	0.95	0.90
Circular Clockwise	0.98	0.98
Circular Anticlockwise	0.99	0.98

Table 5.6.: The precision and recall values are shown.

simple linear movements are more error prone. In general, the permitted gestures can be detected with quite high certainty in real-time. The implementation is simple due to the restricted dimensions of the feature space and can be coded on a simple micro-controller.

5.3. Summary

In this chapter, I mainly deal with the challenge of transferring offline solutions to online real-time applications. This chapter especially addresses the research question 6 *how to scale model complexity used in* ***real-time applications***. I motivated this issue by leveraging two applications with capacitive proximity sensors.

A smartphone, desktop PC, or other cloud infrastructures nowadays already provide a large quantity of integrated computation resources. However, edge computation devices are still restricted by the computation capabilities and the missing storage backbones. Preliminary request is therefore to save computation efforts in order to achieve quasi real-time system response on these devices with limited hardware resources.

In this chapter, I aim at resolving this issue from the model side. In case of the exercise tracking application with the capacitive proximity yoga mat, I mainly benefit from the end-to-end learning by mitigating the step of handcrafting features prior to the classification. Along the processing pipeline for developing a sensor-driven application for HAR, the most time consuming step is to calculate the hand-designed features. By leveraging the end-to-end learning structure from the current deep learning methods, we shifted the feature extraction step inside the optimization network. The inference for a trained model always takes a fixed amount of time despite of the amount of training samples.

In case of the mid-air gesture recognition system with capacitive proximity sensing, I take advantage of the simplicity of the underlying model with only a limited set of gestures. The feature space is strongly restricted by extracting only angular features of position data and the amplitude of the moving vector feature. Dynamic time warping method on univariate symbolic time series is fast compared to multivariate quasi-continuous time series. In addition, model-based approach in the proposed application mitigates the need of carrying data samples to support data-driven model decisions, such as for k nearest neighbour approach or support vector machines where the computation time scales with the amount of data. Thus, model-driven approach further reduces the storage requirement on embedded devices.

6. Stationary systems for a smart environment

In previous chapters, we have individually discussed the main components within the general processing framework for designing a sensor-driven application for HAR tasks in detail. I have linked the main challenges within the framework with examples, applications and experiments. This chapter aims at providing an overview of applying the previous findings to answer the main research question towards *how to setup a successful HAR system* from system design to a working prototype.

In this chapter, as a link to the main research question of the thesis, I propose two stationary systems for smart environment. These systems are stationary applications. Stationary applications target at ambient and installed systems that can be hidden from plane view and make the environment smart. They are integrated into the environment and ubiquitously sense the human perception. This kind of applications are especially suitable for implicit interactions. Opposed to explicit interaction, implicit interactions target at those interactions, that the user is unaware of the sensing system itself, but the sensing system is still all present and tries to continuously interpret the unintentional actions performed by the user.

In order for a smart environment to provide services to its occupants, the need for understanding the current state or context is of critical importance. This ability enables us to build human-centered applications, which aims at making the user's life more comfortable and safer.

As an essential requirement of pervasive computing, free and natural interaction is considered as necessary. This attracts the interest of various researchers. Instead of using distributed sensor networks, such as binary sensors integrated into furniture, or motion sensors and ambient sensors placed on various locations, I am more interested in leveraging only one single ambient sensor system, which provides more precise and high level information than just binary values. Therefore, in this chapter, my focus is on investigating two floor-based applications realized with two different sensor categories (surface acoustic sensing and electrostatic sensing), in the smart home context.

Applications dedicated for smart homes have to fulfill certain requirements, as it is invading the domestic area of individuals. It should preserve the user's privacy above all. This excludes most of the vision-based applications and acoustic speech event based applications. It should be power efficient and with minimalist design efforts to allow edge computing. Thus, my contributions to this research field are grouped as follows:

Passive acoustic sensing

- Contribution 1: Design of tag-free surface acoustic arrays for analyzing human activities of daily living in a smart environment (Section 6.1.1)
- Contribution 2: Investigation of minimum sensor setup for accurate activity recognition (Section 6.1.1)

Electrostatic sensing

- Contribution 1: Scalable floor-based indoor positioning system with a low cost, grid-based sensor system using electric potential sensors (Section 6.2.2)

I will first propose a stationary system build with surface acoustic sensors, which measures only the surface vibration waves caused by environmental forces. This system aims at detecting deterministic and characteristic vibration patterns from ambulation activities such as footsteps or object induced acoustic events. According to these vibration patterns, we are able to draw contextual information used for recognizing a set of activities of daily

living. Starting from a sensor array with four pickups, we further investigate a minimalist sensor arrangement to accurately detect surface vibration induced activities.

The second contribution in this chapter is to introduce a scalable floor-based indoor localization system using electric potential sensing. Advantage of using the electrostatic sensing is due to its power efficiency and large detection range. Grid-based layout and the modular composition further enables us to scale it to different room geometries and spatial resolutions.

6.1. Passive acoustic sensing

Passive acoustic sensor converts mechanical vibration into electric signals. The most common use-case in human computer interaction is the footstep detection and resulting from this, recognition of activities of daily living. Pan et al. [PBJ*14] proposed a room-level building occupancy estimation system by utilizing low-resolution vibration sensors that are sparsely distributed. The goal of their work was to extract features that enable binary classification between actual occupant activities and noise prone ambient vibrations. Applying footstep detection, they are able to track individuals and generate useful trajectories. In their proceeding work [PWQ*15], they further introduced the indoor person identification ability of their system by utilizing the footstep induced structural vibration. Since the structural vibration can be measured without interrupting human activities, they claimed that their system is suitable for many ubiquitous sensing applications.

Micro-Electro-Mechanical Systems (MEMS), such as piezoelectric elements, active microphones or seismic sensors can be used to measure the vibration patterns. Simple piezoelectric thin film can be made using inexpensive materials, but require external amplification. The charge amplifier and other electronics need to be carefully designed and placed as close as possible to the sensor to avoid noise coupling and other signal errors. The mounting process to the sensing surface is also complex. A firm coupling to the sensing surface is required. This is done by putting proper weights on the sensing element to induce certain tension. However, over-tightening can negatively affect the output sensitivity. For simplicity, we decide to use Schaller Oyster pickup sensor for broadband signal reception with no external amplification requisition. Schaller Oyster 723 [1] is a piezoelectric pickup. An excellent signal reproduction is achieved through a combination of a specially designed membrane, a piezoelectric element and a plastic contact gel to secure a good coupling to the surface. It is a transducer, that captures or senses mechanical vibrations and convert this signal to an electrical signal that is amplified with an internal instrument amplifier. This electrical signal is used to generate deterministic vibration patterns from different event sources. Thus this system provides a compact solution for rapid prototyping without the need to self designing external hardware circuits.

In the scope of our research work, we intend to discriminate more fine-grained activities of daily living (ADL). Apart from the main level of activities such as recognizing footsteps, falls and furniture usage, we further consider a finer determination of the sub-activities. We extend the general footsteps into walking bare foot, high-heels or with shoes in general. Further we distinguish the vibration events of the furniture according to its localization in the sensing areas. This fine resolution enables us to design better interaction interfaces. Beyond the ADL recognition, we also investigate the room-level coverage with a sparsely distributed sensor array setup as well as the accurate classification with minimum sensor coverage. This section of the thesis is based on our published work in [FMK*18].

[1] Schaller Oyster External pickup, url = https://www.acousticcentre.com.au/products/schaller-oyster-external-pickup, last accessed on 2020-04-08

Figure 6.1.: Figure depicts the hardware used. The M-Audio M-Track Quad is depicted in the left panel, which can connect up to four input channels. In the right panel of the Figure, one of the four Schaller Oyster 723 pickup sensor used in our experiment is depicted.

6.1.1. Physical principles of acoustic surface wave

Applying force on a surface, for example through footsteps or falling creates an acoustic event. This can be described physically as a mechanical bump. It lets the surface vibrate and a mechanical wave propagates through the surface and transmitted to the surrounding air, resulting in secondary acoustic waves as explained in [JGC80]. The particle movement is normal to the surface and parallel to the direction of propagation. The surface acoustic wave is called the Rayleigh wave. This mechanical wave transmitted through the surface can be sensed and sampled with a passive piezoelectric pickup sensor.

The sampling frequency in our application is 96 kHz, as this is the maximum supported sampling frequency of the M-Audio M-Track Quad audio device used in this project. The higher the sampling frequency, the more information can be reconstructed due to the Nyquist-Shannon sampling theorem as explained in [Wei01]. It states that we can reconstruct a signal with the fastest frequency component up to 48 kHz without information loss for our chosen sampling frequency. This is especially important for sudden events like falls or steps which contain high-frequency components in the signal itself. According to the high sampling frequency, the time resolution increases. With the sampling frequency of 96 kHz, we achieve a time resolution of 42.67 ms for a window length of 4096 samples and the time resolution decreases to 85.3 ms with an increased window length of 8096 samples. The given windows size is used to extract features for activity classification. We use the sliding window approach with an overlap of 50 % between successive window frames.

The M-Audio M-Track Quad[2] with four connected Schaller Oyster 723 Pickups is utilized for the pipeline interface and the dataset recording. Figure 6.1 shows this hardware. The M-Audio M-Track Quad is an audio input and output device for recording and playing music. It is possible to add additional signal processing hardware. One advantage of this device is that it can handle four independent microphones. On-board sound cards from laptops or computers support only one or maximal two channels. Every microphone input channel has its own amplifier control and can be set to the instrumental or microphone input handling. This device can be connected directly to the computer and handle four channels with the ASIO driver under Microsoft Windows.

[2]M-Audio M-Track Quad USB Audio Interface, url = https://www.musicstore.de/de_DE/EUR/M-Audio-M-Track-Quad-USB-Audio-Interface/art-PCM0012168-000, last accessed on 2020-04-08

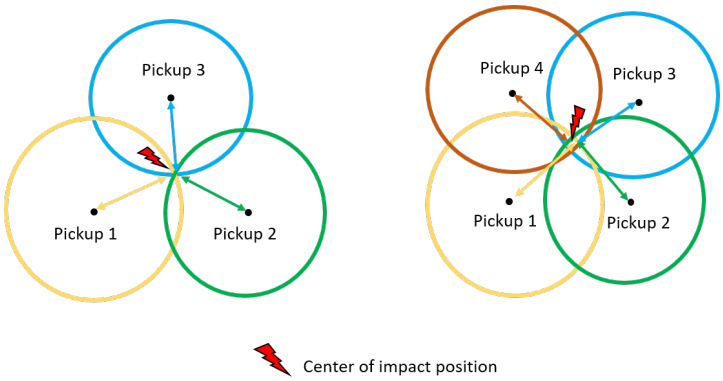


Figure 6.2.: *Left*: The concept of Tri-/Multilateration on the ground plane is depicted here. *Right*: Due to input noise, there is often no ideal solution for the TDOA. Therefore a least square approach is used to minimize the distance error towards the receiving sensors.

Localization with Time Difference of Arrival The four sensors setup is used to provide location information within the sensing area. A general method used for localization in the time domain is the time difference of arrival (TDOA), as used in [LSWL12]. By measuring the time delay τ_{ij} of the same source of impact arriving the sensor pair i and j, we can setup the following Equation (6.1),

$$D_1 - D_2 = \tau_{12} \cdot v = ||\mathbf{x}_1 - \mathbf{x}_c|| - ||\mathbf{x}_2 - \mathbf{x}_c|| = \sqrt{(x_1 - x_c)^2 + (y_1 - y_c)^2} - \sqrt{(x_2 - x_c)^2 + (y_2 - y_c)^2} \tag{6.1}$$

where τ_{12} is the time difference between the pickup 1 and pickup 2, the $\mathbf{x}_1 = (x_1, y_1)$ and $\mathbf{x}_2 = (x_2, y_2)$ represent the center position of the pickup sensors, and $\mathbf{x}_c = (x_c, y_c)$ indicates the center position of force impact. By integrating another pair of pickup sensors, we can theoretically calculate the exact force of impact using the Equation (6.2).

$$\tau_{13} \cdot v = \sqrt{(x_1 - x_c)^2 + (y_1 - y_c)^2} - \sqrt{(x_3 - x_c)^2 + (y_3 - y_c)^2} \tag{6.2}$$

By assuming the same material structure, the speed of propagation v are assumed to be homogeneous and equal in all directions. Therefore the term v contributes equally in all equations.

The time difference τ_{ij} of the individual receiver pairs can be calculated from the received signals by leveraging signal correlation to find the optimum peak positions. Theoretically, an exact solution of the impact position can be exactly defined by using only three receivers as depicted in the left panel of Figure 6.2. In practice due to noisy measurement in the real-world application, it is not always possible to get the exact solution of the problem. Therefore we use an additional receiver to perform least square solutions, where we try to get a solution with minimum squared errors with respect to all receivers. This case is depicted in the right panel of Figure 6.2. By leveraging the TDOA approach, the quasi exact position of the point of impact can be determined. Since the force of impact is directly transferred to the ground, there is no z component.

6.1.2. Vibration detection of human activities in smart environments

Now we will introduce basic feature extraction methods used for activity recognition with time series. For time signals, there are basically three feature domains which are commonly used. They are either features from

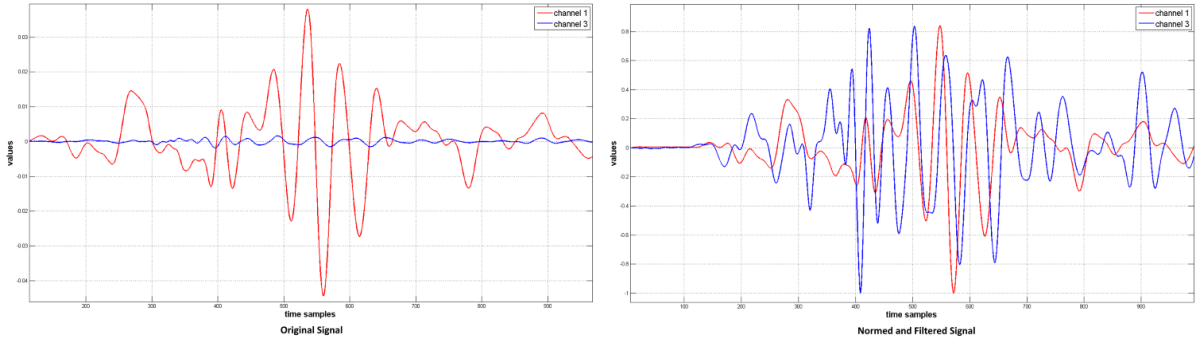


Figure 6.3.: Depicted are the input signals from channel 1 (red) and channel 3 (blue). The left figure shows the unnormalized signals and the right figure shows the normalized signals. Both signals are more similar as they appeared to be in the raw format relate to signal amplitude and signal appearance. The exact axis labeling is obsolete, as these figures aim at indicating the effect of applying normalization on both channels. Normalization mitigates the issue of hardware dependent channel attenuation.

time domain, frequency domain or time-frequency domain. In the time domain, we investigate the pure signal appearance and extract features such as zero-crossing or amplitude related features. In the frequency domain Fast Fourier Transformation (FFT) [RKH10] is often applied to get the spectral information of the signal with respect to certain frequency bin. In time-frequency domain we are interested in the spectral distribution of the signal over discrete time steps. This can be done using short time Fourier transformation (STFT) [Grö01].

We perform normalization and synchronization as part of the data pre-processing, to achieve a better classification result. By observing different input signals from all input channels, we identify a systematic offset in time from different input channels due to the synchronization issue of the device hardware. Since the error is systematic, we can remove this offset. Peak-to-Peak normalization is applied to mitigate the hardware dependent attenuation effects from the input channels. Since the amplitude is different for each input channel, it is of utmost importance to normalize the input signal before applying classification approaches by converting different channels to the same magnitude range. This effect is illustrated in Figure 6.3. Before normalization, the signal from channel 1 seems intuitively different from the signal recorded from channel 3. After applying the normalization step, both signals appear more similar to each other. This improves the correlation result of finding signal delay component in the localization step.

A list of the handcrafted features extracted is given below. Besides the location of impact, we use these features as input per sliding window to evaluate the different classifiers, such as support vector machine (SVM), classification and regression tree (CART) approach, and Bayesian decision network (BayesNet):

1. RMS value
2. Zero-Crossing value
3. The maximal FFT value index
4. The FFT value average
5. The FFT Standard Deviation
6. The FFT vector

The first two features are both extracted from the time-domain. The RMS value can be calculated by using the Equation (6.3). x_i is the sampled time signal, and N is the number of samples within a chosen window size of

4096 or 8192 samples.

$$RMS = \sqrt{\frac{\sum_{i=1}^{N} x_i^2}{N}} \tag{6.3}$$

Zero crossing counts the number of zero crossings within a chosen window of N samples.

The FFT gives the spectral distribution of a time signal in the frequency domain. The Equation (6.4) calculates the Fourier coefficients. $x(n)$ are the discrete sampled time signal and N is the number of FFT points used. $(k = 0, 1, ...N-1)$ represents the index of the Fourier coefficients.

$$X(k) = \sum_{n=0}^{N-1} x(n) \cdot \exp(-j\frac{2\pi}{N}nk) \tag{6.4}$$

The maximal FFT value index represents the strongest spectral power at a certain frequency and the average FFT value is calculated by $\bar{X} = \frac{\sum_{k=0}^{N-1} X(k)}{N}$ and its standard deviation is $s = \sqrt{\frac{\sum_{k=0}^{N-1} (X(k)-\bar{X})^2}{N-1}}$. The FFT vector consists of components $X(k), k = 0, 1, ..N-1$ which are the Fourier coefficients.

For extracting handcrafted features, the design choice are relying on the inductive biases from domain expert knowledge. These data-driven knowledge were derived from carefully investigating the input time signals during the data processing stage. Examples of the time domain signals for various target classes are depicted in Figure 6.4. As can be observed from the signal appearances, different activities have its own deterministic signal patterns. Certain activities, such as *fall* and *high heels* both containing more high frequent components compared to other classes, such as the class of *walking bare foot*.

6.1.3. Experimental setups and evaluation

In order to evaluate our proposed system in the smart home area, four pickups are placed in a room with a parquet floor with the size of 3.9 x 3.42 meters. We chose solid surface material over soft carpet to ensure a sound ground coupling of the vibration signal induced by the acoustic events. In this room, there are various types of furniture (e.g., cupboards, tables and sideboard). The setup is illustrated in Figure 6.5. Table 6.1 lists the positions of specific entities. For the evaluation setup, a wide range and sparsely distributed pickup sensor setup is used. The advantage of this setup type is the larger observation area. The amplifier of the M-Audio M-Track Quad is set to the maximal level to also detect weak signal strength activities such as *Walking barefoot* and to reduce missing signals. After setting up the environment, the activity signals from the pickups are recorded. With the recorded data an offline analysis with the WEKA Explorer [HFH*09] is performed. WEKA provides an analytic tool for accessing and evaluating machine learning algorithms and approaches.

The evaluation starts with an analysis of the activity groups. We start with a basic activity group (AG-1), including only *walking*, *cupboard closing* and *falling*. We then try to expand this basic activity groups into various sub-activity groups including more precise sub-activities. In AG-2, we divide the class of *walking* further into a more subtle distinction: *walking barefoot*, *walking with shoes* and *walking with High-Heels*. In AG-3, we expand the class of *cupboard closing* to 3 cupboards locating at different positions in the room. In the final activity group of AG-4 we combine all possible activities to verify the overall performance of the recognition. For better clarification, the different setups are listed in Table. 6.2.

We conduct the evaluation study with 8 participants to perform the described activities in the physical setup as shown in Figure 6.5. The test population consists of 3 females and 5 males, with an average age of 35 years old and body weight ranges from 65 kg - 130 kg. There are no specific instructions given as how to perform these activities. The door of the cupboard can only be opened in one direction. Only the closing of the cupboard can be

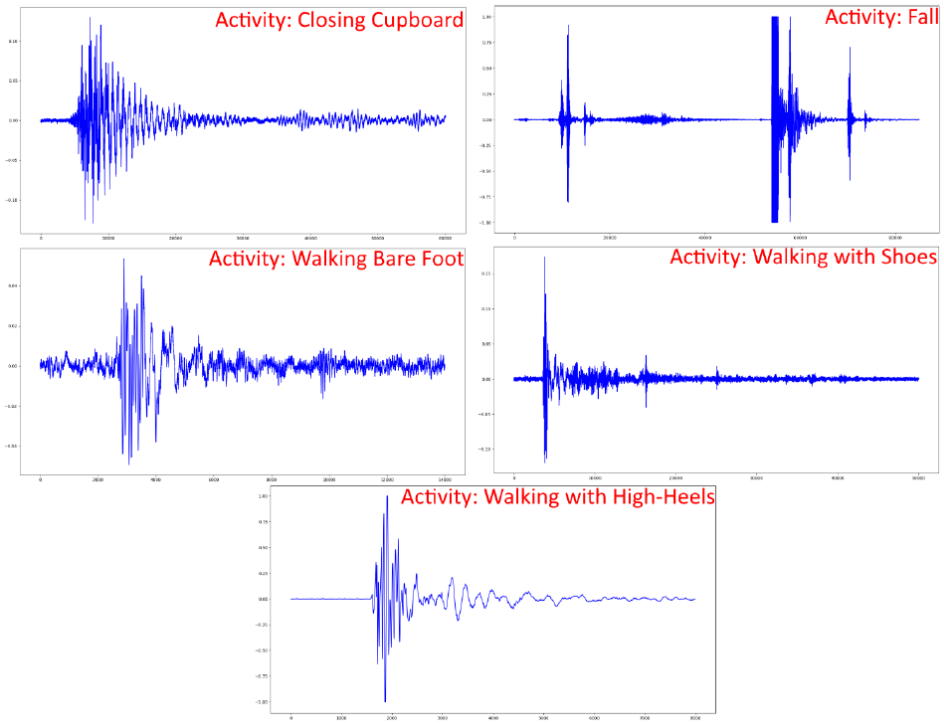


Figure 6.4.: Time-domain signals for the specified activities are depicted. Certain activity classes, such as *fall* and *high heels* possess higher frequent components than the class of *walking bare foot*. (The exact axis labeling is obsolete.)

Object	Position x	Position y
Pickup 1	0.32 m	0.37 m
Pickup 2	3.22 m	0.65 m
Pickup 3	3.81 m	3.26 m
Pickup 4	0.49 m	3.27 m
Cupboard 1	0.90 m	0.30 m
Cupboard 2	2.52 m	0.30 m
Cupboard 3	3.60 m	0.90 m
Fall Area	2.20 m	1.30 m

Table 6.1.: The positions of specific entities in the evaluation environment.

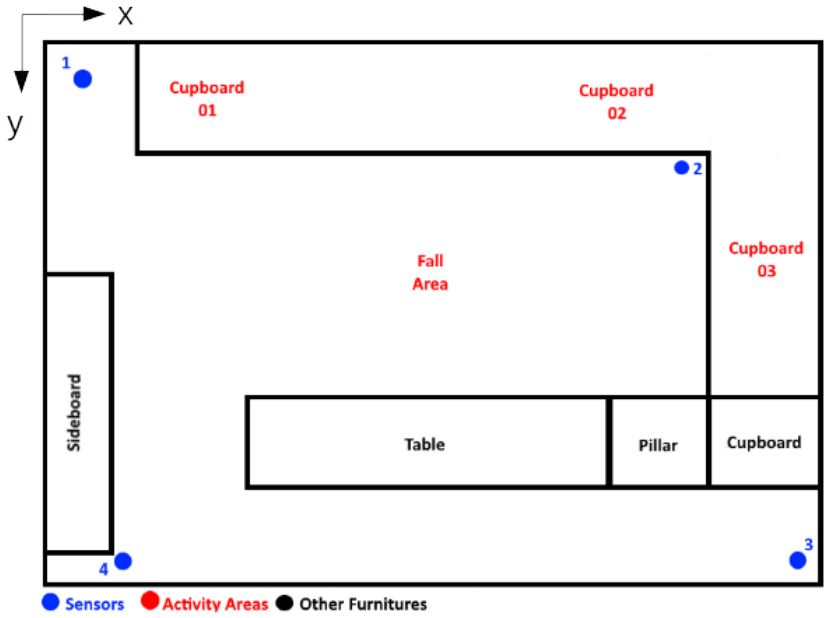


Figure 6.5.: Setup of the evaluation environment. The experimental setup is the kitchen area in a living laboratory with wooden floor tiles to ensure a good coupling and propagation of vibrations caused by force impact.

AG-1	**AG-2**	**AG-3**	**AG-4**
Walking	Bare Foot	Walking	Bare Foot
-	Shoes	-	Shoes
-	High-Heels	-	High-Heels
Cupboard	Cupboard	Cupboard-01	Cupboard-01
-	-	Cupboard-02	Cupboard-02
-	-	Cupboard-03	Cupboard-03
Falling	Falling	Falling	Falling

Table 6.2.: It depicts the four different experimental setups. In subsequent setups, the classes are refined to distinguish more similar sub-classes.

registered by generating a vibration event. We collected in total 400 steps, (including bare foot signals, wearing shoes or high heels) and 400 cupboard closing events from three cupboards located in different places. *Falling* is a special case, because it is hard to simulate. A protection mat is placed on the ground to catch up the force, but will distort the signal since it dampens the vibration event. Therefore, the fall simulation is carefully performed and not too often on the hard floor to prevent injuries. Therefore we only collected 50 times of falls. So the result feature vectors are up-sampled multiple times to match the number of vectors of the other activities in the training dataset to avoid over-fitting. This way we can modify the dataset that we use to build our predictive model to have more balanced data classes. The labelling of the different activities are done by the participants on their own.

In general, every impact event starts with a significant amplitude rise and followed by a fast amplitude fall. The amount of time in which an impact event takes place is very short. Experiments show, that the average event length in time took approximated between 8000 to 12000 samples, corresponding to a time window of 83 ms to 125 ms with a sample frequency of 96 kHz. Thus the training dataset, is segmented with the sliding window approach using a sample window length of 4096 or 8192 samples and an overlap of 50 %, contains now 1000 samples each class. Due to this limited amount of data samples and the discriminative features in the time series classes, we limited our classifiers to the most prominent conventional methods.

With this amount, the experiment results are considered to be sufficient for conventional classifiers. Each classification experiment with WEKA is performed with a window size of 4096 and 8192 samples and in each case with normalized windowed signals. To evaluate the performance of different classifier methods, statistical measures like precision and area under curve (AUC) are given. The precision also known as positive predicted value gives a measure of the portion of correctly predicted class samples with respect to all predicted positive classes. The AUC is the integral of the receiver operation characteristic (ROC) curve and presents the performance of a binary classifier. The ROC curve is a plot of true positive rate (TPR) against false positive rate (FPR). It shows the performance of this binary classifier with respect to false rejection and false acceptance. An AUC value of 0.5 is close to a binary classifier with a random guess. The closer the AUC gets to 1, the better this binary classifier works.

Figure 6.6 (**a**) shows the precision of different activity groups as presented in Table 6.2. The window size of 4096 samples is used and the time signal normalization is applied before feature extraction to reduce the channel difference. SVM, CART, and BayesNET are used as classifiers with 10-fold cross-validation to show the more generalized performance. We use the Bayes inference net as the baseline classifier, since it is a fairly simple model. CART with binary decision trees is used to better generalize the model. SVM is used in combination with grid search combining cross-validation approach to find the optimum hyper parameters. We used the radial basis function (rbf) kernel to separate the sample classes. The regularization parameter C and γ is optimized by grid search. With respect to the training data, the final hyper parameter set is chosen such, that $C = 10$ and $\gamma = 10^{-3}$ are used as regularization parameter to avoid the classifier from overfitting.

The classification of the three general classes in AG-1 can be achieved with a precision of more than 95 %. The SVM shows superior performance in all classification results than CART and BayesNET. The SVM reaches the precision of 98.47 % for AG-1, 94.26 % for AG-2, 96.50 % for AG-3 and 93.61 % for AG-4. As expected, the precision decreases as the number of sub-classes increases, since the sub-activities are not fully independent from their main supper classes. This similarity decreases the separability of the SVM classifier and makes a clear separation without overfitting more difficult. The AUC for each classifier and each activity group can be extracted from Figure 6.6 (**b**). The SVM shows more superior classification performance in all four setups regarding the higher AUC value.

The precision can be however increased by using a larger window size of 8192 samples as shown in Fig 6.6 (**c**). The size of the samples within a window is of importance to the performance of a good classification. It

w=8192, norm FFT			
	Cupboard	Falling	Walking
Cupboard	**977**	3	20
Falling	0	**1000**	0
Walking	5	5	**990**

Table 6.3.: Cross Validation Confusion Matrix for SVM classification of AG-1.

determines how well one activity is observed within the observation window. All features are inferred from the chosen observation window. The final results of all activity groups are superior to the case of 4096 samples, because with increasing window size we can observe a full acoustic event, especially in case of a strong similarity between classes. The SVM reaches the precision of 98.90 % for AG-1, 97.42 % for AG-2, 98.42 % for AG-3 and 97.23 % for AG-4. The AUC for individual classifier and activity groups for this window size can be extracted from Figure 6.6 (**d**).

6.1.4. Minimal sensor setting for accurate human activity recognition

We further investigate the performance of using only one single input channel instead of using a combination of 4 pickups. We state the hypothesis that by using only one single pickup, the performance is inferior than leveraging the fusion of all 4 pickups. However, it is interesting to examine, if it is possible to realize reduced computation with the cost of slightly decreased performance. The result can be seen in Figure 6.7. Since from previous results the SVM performs the best, we keep using SVM as the best practice classifier to validate the hypothesis. Even the precision of using only one input channel is not worse in certain cases, the fusion still outperforms the single input case. The result indicates that the overall performance of different channel inputs is not stable. Certain input channels perform better compared to others according to the sensor location and hardware property. In addition, by using only one single input, we are missing the location information of the source of impact by applying the TDOA approach.

Coming back to the full setup, the confusion matrix for SVM classifier with 10-fold cross validation is presented in Table 6.3 for AG-1, Table 6.4 for AG-2, Table 6.5 for AG-3 and Table 6.6 for AG-4. Confusion matrix indicates the performance of a classifier by providing an overview of positive and negative classification for each class in a single table.

The strong diagonal elements indicate the good performance of the SVM classifier for almost all classes. Especially the class **falling** is detectable with almost 100 % certainty through all the different activity groups. In the extended activity use-cases, we observe that the misclassification of different sub-activities on the off-diagonal elements increases. It is not unexpected to notice that certain sub-activities like *walking bare foot* or *walking with shoes* are getting confused sometimes, since they are not fully independent.

6.1.5. Discussion of passive acoustic sensing

In this section, we proposed a ubiquitous and unobtrusive stationary floor-bounded system using four pickups to recognize fine-grained activities of daily living. The system is easy to install, has moderate power consumption and is easy to maintain. The use of pickups has the further advantage that only acoustic waves of the surface are recorded and any environmental sounds or natural speech will not be directly recognizable. The computation effort is rather low compared to image processing and does not suffer from the problem of occlusion.

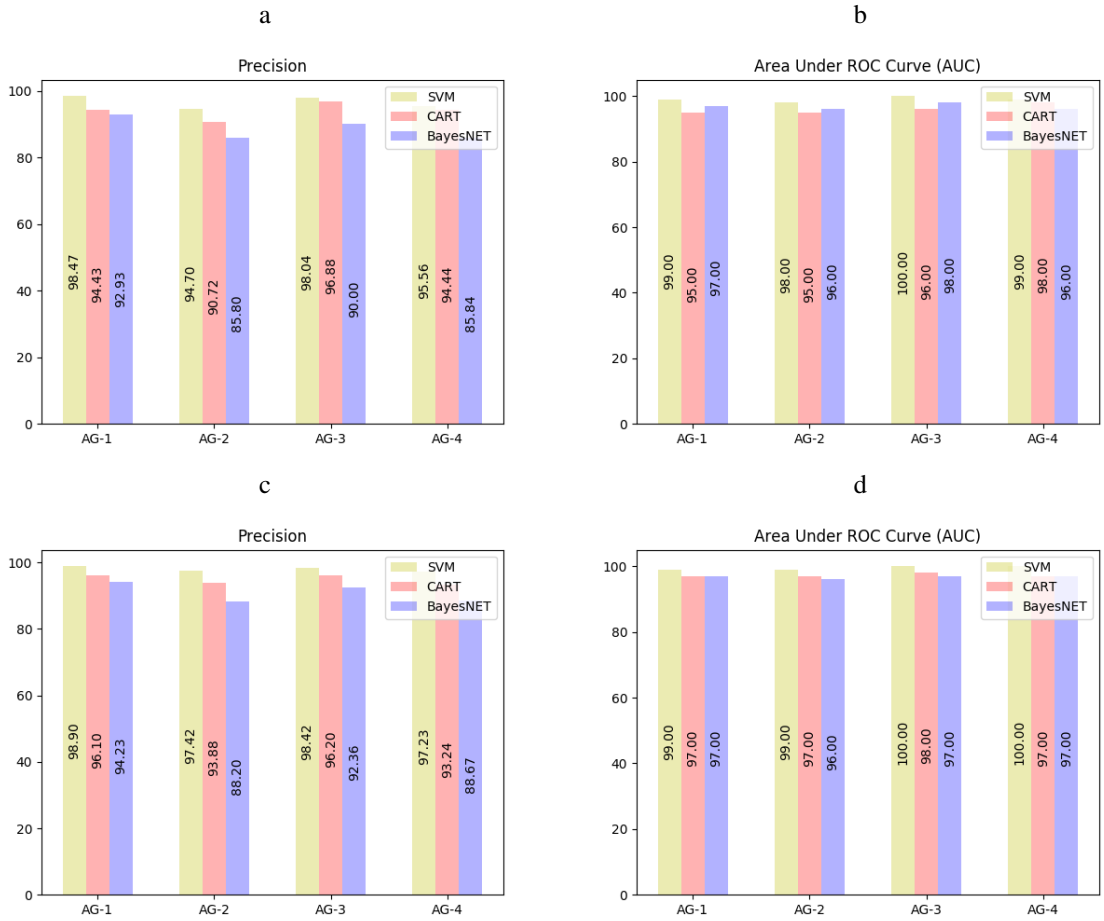


Figure 6.6.: Figure **a**: Precision score for different activity groups with a window size of 4096 samples. The SVM classifier uses a regularization parameter C=10. Figure **b** shows the ROC Area combine for different activity groups. Figure **c**: Precision score for different activity groups with a window size of 8192 samples. The SVM classifier uses a regularization parameter C=10. Figure **d** shows the ROC Area combine for different activity groups.

w=8192, norm FFT

	Cupboard	Falling	Bare Foot	Shoes	High Heels
Cupboard	**975**	2	14	9	0
Falling	0	**1000**	0	0	0
Bare Foot	2	2	**926**	60	10
Shoes	0	2	14	**980**	4
High Heels	1	1	1	7	**990**

Table 6.4.: Cross Validation Confusion Matrix for SVM classification of AG-2.

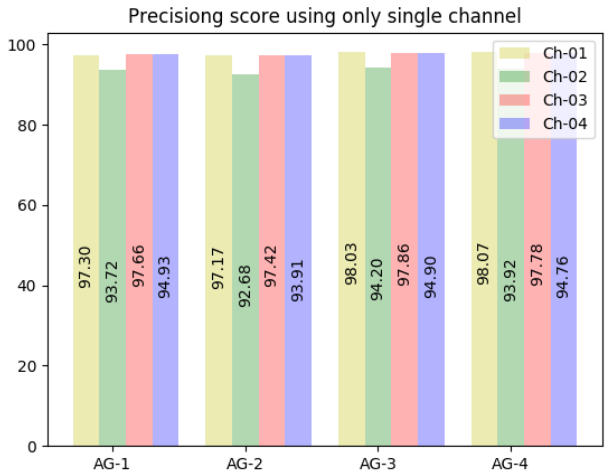


Figure 6.7.: Precision score for different activity groups each using only one single channel. The SVM classifier with a linear kernel is used. The precision score is built using 10-fold cross validation.

w=8192, norm FFT

	Cupboard-01	Cupboard-02	Cupboard-03	Falling	Walking
Cupboard-01	**967**	2	31	0	0
Cupboard-02	0	**1000**	0	0	0
Cupboard-03	36	0	**964**	0	0
Falling	0	0	0	**1000**	0
Walking	2	3	0	5	**990**

Table 6.5.: Cross Validation Confusion Matrix for SVM classification of AG-3.

w=8192, norm FFT

	Cupboard-01	Cupboard-02	Cupboard-03	Falling	Bare Foot	Shoes	High Heels
Cupboard-01	**965**	0	33	0	0	2	0
Cupboard-02	0	**992**	0	0	2	6	0
Cupboard-03	37	0	**959**	0	0	4	0
Falling	0	0	0	**1000**	0	0	0
Bare Foot	0	0	0	2	**925**	62	11
Shoes	0	0	0	0	22	**976**	2
High Heels	0	0	0	3	2	6	**989**

Table 6.6.: Cross Validation Confusion Matrix for SVM classification of AG-4. A window-size of 8192 samples and normalized FFT is used as input features.

We use pickups to retrieve surface acoustic waves to track a basic set of user's daily activities, including *walking*, *cupboard closing* and *fall*. By exploring different algorithms, we achieved a desired precision of over 94 %. Additionally, the activity classes *Walking* and *Cupboard closing* are further expanded into their sub-classes (shoe types and cupboard instances) and can be successfully distinguished with a relatively high precision rate. This goal is achieved by using a fusion of multiple pickups instead of only a single input channel. We have further shown that by using a combination of multiple pickups, the performance is more superior and stable than only using a single one of them.

One limitation of the setup is the connection of the pickups to the ground surface. To perfectly pick up the surface acoustic wave created by the impact, a sound coupling of the sensing device and the ground surface is of utmost importance. It should be noted that the ground surface used in the experiment is homogeneous. In case of an inhomogeneous ground surface, like, e.g., placing a carpet on the wooden floor, a calibration of the system is needed to maintain a high recognition performance. It has been shown in our experiment, that carpet attenuates the acoustic surface wave and decreases the amplitude of the received signal. Since the received signal is relevant to the performance of our system, therefore it is recommended to use a homogeneous flooring material, which can perfectly conduct the surface acoustic wave.

One pitfall of human activity recognition is the data acquisition and labeling phase. First, we have to manually start the recording and then walk back to the specific position of cupboards. Thus, this makes the cupboard events include some footstep events in the beginning. Falling down can contain an initial step or steps after standing up to stop the recording. Therefore the falsely classified samples can be explained by not perfectly dividable class samples. Since the class of High Heels is mainly worn by female participants, it is unbalanced similar to the class of fall. We have to use the Synthetic Minority Over-sampling Technique (SMOTE) [CBHK02] to over-sample this class. However, this synthetic method introduces less overfitting issue compared to simply repeating samples. Comparing the frequency components of the step class, high heels signals contain higher frequency components as expected and makes it more distinguishable in comparison to other footstep samples.

To reduce the computation load, we further discard the non-activity class by using the impact detection step before classification. Avoiding the continuous processing of each window, the heuristic that is outlined by Braun [BKK15] is adapted in our use-case. In order to detect the impact window, we compute the RMS value of each window and defined a MINRMS value to ignore any window with a RMS value that does not exceed this threshold. The window with the maximal RMS value between the first and last RMS value that crosses this threshold is chosen. To avoid choosing a local maximum, all peaks are ignored after selecting the first maximum until the RMS values fall below a CLOSINGRMS value. We further adapt this approach to our multi-channels

application by using the following Equation (6.5),

$$RMS_{window} = \sqrt{\frac{\sum_{c=0}^{N} RMS_c^2}{N}} \tag{6.5}$$

where N is the channel count and RMS_c is the RMS value of the channel c. This method allows us to build a real-time system by minimizing computation efforts and quasi real-time response on selected windows. Since the output of the SVM can be a measure of probability instead of one-hot decision, we can further combine SVM with threshold method to determine the correctness of the classified class.

Finally, the current system only works for single user. Multiple users would affect the time series which leads to modifying the extracted features and therefore decrease the classification performance. For multiple person, independent component analysis (ICA) [HO00] can be applied to isolate the individual contribution to the system. The decomposed individual signal can be further processed to determine the true class of activities.

conclusion Promising classification results show the potential of using such a system for real-time *falling* recognition. Accurate falling detection at home is useful to assist elderly people and prevent bad injuries or outcomes from falling. Two-staged processing further reduces the computation effort by only extracting features from time windows containing real activities.

Though, we achieved a relative high recognition accuracy, this kind of passive vibration sensors strongly relate to the coupling of the sensing surface. The propagation velocity of vibration in materials is strongly dependent on the surface materials. Therefore, such systems are restricted to perform well under constraint conditions. In order to resolve this issue, we further investigated another sensing technology to build another floor-based system for smart homes, which loosens the constraint of sensing surface dependency and allows a more precise indoor positioning.

6.2. Passive electric potential sensing

Instead of using surface vibration sensors to enable room-scale detection of human activities, in this section, we consider the possibility of installing a grid-layout of electrostatic sensors under a non-conductive floor tiles to perform indoor localization. The objective is to reduce the coupling issue of the vibration sensing devices for a floor-based application with a fixed installation.

An accurate indoor positioning system enables location-based services or the controlling of smart home appliances. For example, such a positioning system can be applied in building occupancy detection [FOL16], energy conservation in smart living context [LBG11], or elderly care. Common localization systems can either be token-free or token-based. The latter include systems such as RFID [SN11], ultrasonic sensor arrays [HN10], or WiFi signal based systems [LWNS07], which require the tracked object to actively carry a token. Token-free systems include camera systems such as introduced by Williams et al. [WGH07b] or capacitive sensing smart floors, as proposed by Steinhage et al. [STS*13], Braun et al. [BHW11] and Valtonen et al. [VMV09]. These systems do not require the user to carry any additional hardware. In these cases, all the information about the location and state of the individual must be extracted from sensing devices embedded in the environment.

Kirchbuchner et al. [KGPH*15] showed in their survey, that experience with Ambient Intelligence increases technology acceptance and reduces fears regarding privacy violations and insufficient system reliability. While participants generally tolerate the monitoring of activities in their home, including bathrooms, they do not accept commercial service providers as data recipients. The authors compare four exemplary systems and show

that camera-based solutions are perceived with much greater fears than wearable emergency solutions. Burglary detection was considered as important as health features, whereas living comfort features were considered less useful. Therefore instead of using camera-based systems, we propose a non-visual, tag-free floor-based localization application.

The advantages of floor-based localization systems are two folds. First, one can build tag-free tracking system embedded for a room-scale sensing area. Pressure-based sensing floor or capacitive proximity sensing systems can be used to track the users without an additional tag worn on the body. This system property is especially preferred to the target group of the elderly, because surveys indicate that elderly occupants with memory impairment are more likely to forget wearing tokens. Second, compared to vision based installations, floor-based installation does not suffer from the problem of occlusion. Therefore, the focus of my research interest is primary on tag-free accurate indoor positioning system with non-visual input.

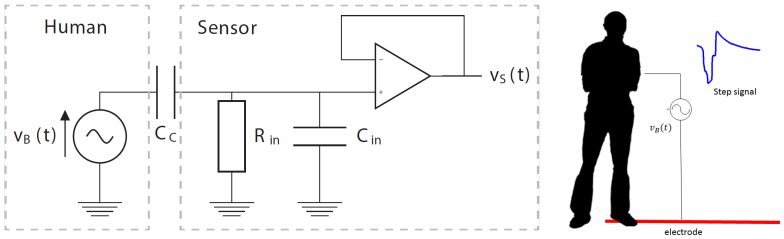
The sensing principle used here is the electrostatic sensing. Opposed to capacitive systems, our system acts to dynamic signals instead of stationary signals. The physical working principle is introduced in the next section.

6.2.1. Physical sensing principle of electric potential sensing

In this section, we shortly describe the physical model of how a passive sensing electrode is affected by a person walking in close range. We leverage ambient electric field distortions that occur due to the presence or motion of a human body. Although the concept of measuring electric field distribution is not new, the use-case and the measurement method proposed by our system is, to our best knowledge, novel. The electric potential sensing (EPS) based smart floor benefits from the fact that each object, carrying a charge, emits an electric field.

During walking or movements in general, every human being is subject to charge accumulation as described by the triboelectric effect in [Fic06]. This changes the capacitive coupling to the environment. Changes in accumulated charge and capacitive coupling result in the fact that a person emits an electric field caused by the varying electric potential in the person's body. We can immediately experience this electric potential when rubbing our hair with a balloon. During everyday activities, this effect is less obvious but still present. To perceive such fields, we measure the induced charge on a sensing electrode in proximity to the field source, i.e. the human person. Every time a person lifts up or sets down a foot close to the measuring electrode while walking, it will cause a charge redistribution on the surface of the sole, which will induce an opposite current on the remote measuring electrode. Also, with varying distance, the capacitive coupling between foot and ground changes while walking, which also affects the body voltage. The relationship between the induced charge Q, the capacitive coupling between the person and the floor denoted as C_c, and the body electric potential difference is thus defined by $v_B = \frac{Q}{C_c}$. The equation for the capacitive coupling is given by the plate capacitor equation, $C_c = \varepsilon_0 \varepsilon_r \frac{A}{d}$, where A defines the sole area and d represents the foot-to-floor distance. Thus, the coupling capacitance increases with decreasing distance of the foot to the sensing electrode.

The capacitance C_c is typically in the order of $0.1 - 10\,\text{pF}$ [FM11]. This weak capacitive coupling requires a very high input impedance to reliably detect the minor displacement current generated by the body movement. In general, the resistance is in the order of $10^{12} - 10^{15}\Omega$ to keep the output voltage v_s stable. Thus to address this problem, we use an impedance converter as depicted in Figure 6.8. Its input has a high impedance so that those very small displacement currents can still be sensed reliably. The sensing electrodes are affected not only by changes in the electric field, which are induced by human activity, but also by the power lines and other electrical appliances in the vicinity. We resolve this issue by low-pass filtering the signal with a cut-off frequency of 15 Hz. This enables us to reliably detect walking activities while suppressing most environmental noises or other electric appliances around 50/60 Hz.



(a) (b)

Figure 6.8.: (a) Impedance converter is required to measure the weak capacitive coupling C_c. The first part represents a human body with a varying body voltage v_B due to movement, while the coupling capacitance is given by C_c. The resistance R_{in} and capacitance C_{in} affect the cutoff frequency of the hardware low-pass filter. The sensed voltage v_s is the actual value measured by our sensor. (b) A step signal of a person stepping on the electrode is illustrated.

To illustrate how a typical sensor signal looks like, we depicted the time series of several footsteps in Figure 6.9. This signal is caused by a person, who is periodically stepping over the measuring electrode as indicated by Label 1 and returning from the other side as indicated by Label 2. Notice that we lift our reference signal to half of the reference voltage, so that in neutral or non-activated situations, the voltage stays a positive constant value and that change of body potentials while walking does not cause a sign flip. Figure 6.9 illustrates the process of a footstep. As soon as the person lifts the foot, the voltage decreases, and increases again at the moment the foot contacts the floor. The same procedure can be observed when we step over the sensing electrode from the opposite side. As soon as a person stands still without any movement, the electrode will discharge to a constant voltage and remain there until new movement occurs. However, it is to notice that a typical step signal appearance could also be reversed, depending on the current electric potential of the person.

The advantage of passive electric potential sensing opposed to active capacitive measurement is that it is purely passive. Electric signal is only induced by moving charges due to dynamic body movement. However, this also poses a major challenge for retaining a signal, if no dynamic change is measurable. Capacitive proximity sensor can be used to measure stationary signals, such the signal remains constant even the person is standing still. One possible solution for EPS system is to retain the last known position, if no further signal is measurable.

6.2.2. Tag-free indoor localization with electric potential sensors

In this section, we proposed the tag-free indoor localization system built with electric potential sensors. The following sections contain extracts from our work published in [FKvW*18]. The structure is as follows: we first introduce the system architecture of our proposed localization application. The algorithm for the determination of the indoor position is then explained in details according to the overall process pipeline given in Figure 6.12. The performance of the proposed application is evaluated on a preliminary study conducted in a laboratory setup. Positioning accuracy of our proposed system is shown for different foot-wears. The significance of the results are shown by performing statistic significance test. It then followed by a thorough discussion on the system limitations and a conclusion with final remarks.

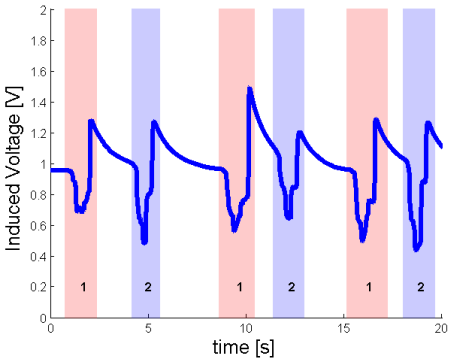


Figure 6.9.: A typical step signal measured by our sensing electrode is illustrated. We measure the change of body voltage induced on the sensing electrode via walking. Label 1 indicates a step forward towards the sensing electrode and Label 2 indicates a step towards to the sensing electrode from the other side.

6.2.2.1. System Architecture

The developed system is divided into three parts: a sensor array on the ground, a sub-controller for each ground segment and a central control unit for data collection and data analysis. On the square floor, electrodes starting from two adjacent sides build a grid structure. The electrodes consist of insulated copper wires and, thus, are very low cost. The wire spacing of 20 cm ensures that an average footprint is reliably detected. According to Ibara et al. [IKH13], a typical stripe length is of maximum 80 cm. Thus, the spatial resolution of 20 cm can be considered as sufficient. The wires are laid out manually, which still entails relatively high labor costs. For a potential market introduction, a laying with roll-out mats should be examined. The sensors are attached to two edges of the bottom surface. Therefore, a replacement of the sensors in case of failure or malfunction is possible without opening the floor surface. This also results in low maintenance costs opposed to other smart floor systems introduced earlier. Our measurement method using EPS ensures that it is possible to place any non-conductive floor covering the electrodes. It is also feasible to embed the electrodes in the screed. In our living lab, we placed a simple carpet above the wire mesh of our test system.

Electrode wires covering certain spatial areas or spaces is called floorspaces or segments. For each floorspace or segment, the sensors are connected via an RS485 driven bus to a sub-controller. This unit manages the communication between the sensors and collects the sensor values with an update frequency of 10 Hz. Each sub-controller can manage up to 64 sensors. Subsystems can be used to cover rooms or single regions of interest in an apartment. Up to eight subsystems are connected via a CAT7 cabling to a central control unit. This main controller consists of a peripheral board connected to a BeagleBone embedded computer, which evaluates the collected sensor signals. The BeagleBone Black is a low-cost, community-supported platform, with an AM335x 1 GHz ARM Cortex-A8 equipped with 512 MB DDR3 RAM and 4 GB 8-bit eMMC onboard flash storage. On top of a customized Linux distribution, we are running Apache Karaf, a Java application container. This enables a modular software architecture which allows successive updates of all components and remote management of the whole system. For each apartment, only one main controller is needed. This ensures that the data analysis is performed locally for the sake of data security. If necessary, the system can communicate over the network with a home control or an emergency call system. The overall system layout is depicted in Figure 6.10.

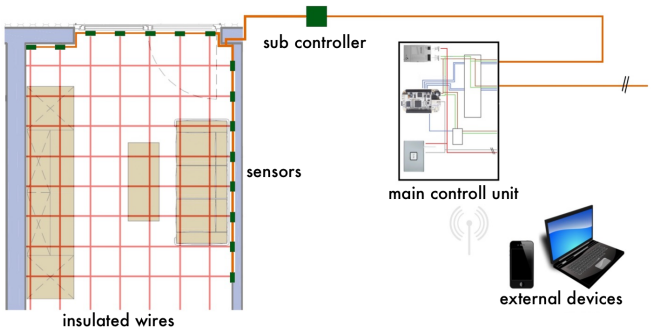


Figure 6.10.: The system overview shows the three parts of the system: sensors, a sub-controller, and a main controller. Sensor values are forwarded to the sub-controller and will be pulled from the main control unit for further processing with a fixed sampling rate. Electrode wires covering certain spatial areas or spaces build floorspaces or segments.

We use a stand-alone planning and evaluation tool for the modeling of the ground surface (Figure 6.11). With this tool a building plan can be parsed automatically. Afterwards the sensor positions and the connected electrodes can be configured by drag and drop. For each electrode, the start and end positions are stored. In addition, regions of interest such as entrances, different floor areas and apartment segments can be marked. For this purpose, the edge points of the surface are defined. These defined rooms are of utmost importance for the control of location-based Smart Home applications.

The information is stored in a hierarchical model as an XML-File on the main controller unit. The model provides the necessary geometric information to our indoor localization algorithm. Figure 6.11 visualizes the system installed in our living lab, consisting of two floorspaces, covering the upper and lower area and one entrance/exit are.

6.2.2.2. Indoor Localization Method

The algorithm used to perform the indoor localization benefits from the model-based structure of the EPS-based smart floor. Using the designer tool introduced in the system architecture, our system is aware of the geometrical distribution of sensing electrodes within different floorspaces, e.g., the bedroom, kitchen, and entrance area. A floorspace is a ground segment defined in the tool according to its geographical location and is, thus, important to enabling smart home services. The process of localization itself is divided into four steps, which will be introduced and detailed in the following section.

The overall process of our proposed indoor localization algorithm can be seen in Figure 6.12. Shortly explained, before we go into details. We start with extracting positions of activities on each activated antennas. One or various region of activities will be further constructed within all floorspaces. The center of each region of interest (ROI) is calculated using the weighted average method. These centers are caused by movement of a person within the measurement area and should then be tracked and updated to form the new current location. Previous location helps to assess the current received measurements.

Activities on Antennas The body electric field changes due to movement. That means, as soon as the foot takes off the ground, we can measure the signal it causes on the underlying electrode wire. We set a threshold to distinguish the true activity from environmental and electrical noise that also has major influences on the

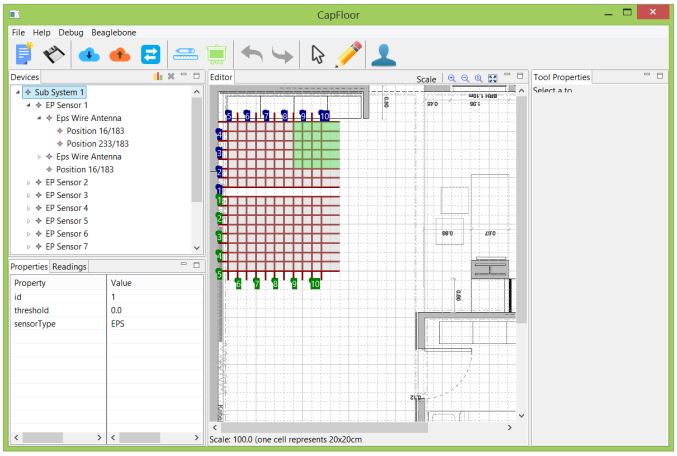


Figure 6.11.: The planning tool supports the installation of a floor. The setup used in our living lab is shown. The green and blue nodes indicate the sensor nodes displaced and the electric sensing wires are drawn in red. The two areas in gray indicate the upper and lower floorspaces and the area in green indicates the entrance/exit area.

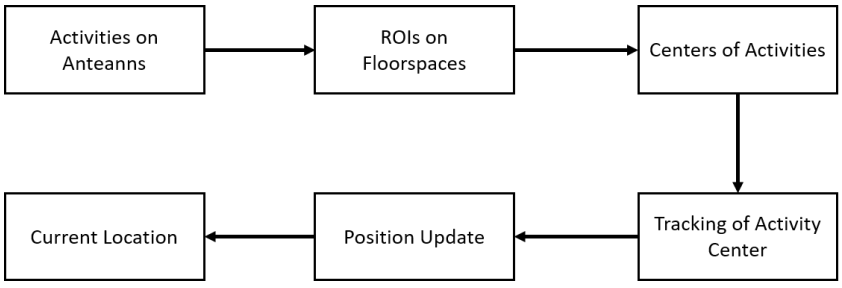


Figure 6.12.: The overall process of our proposed indoor localization algorithm is illustrated.

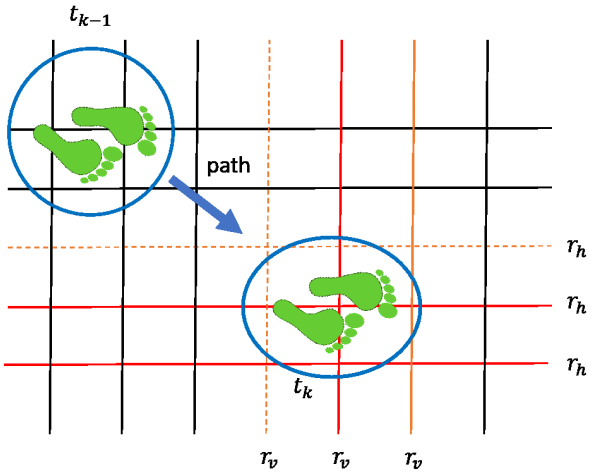


Figure 6.13.: The area of impact is drawn in blue circles according to the step taken. This is strongly dependent on the change of body potential field according to the step. Signal value fades to the edges from the center of a step, which is indicated by a strong red color and fades gradually to brighter red tone.

measurement. Power line noise from electronic appliances generate electric noise component at a frequency of 50/60 Hz. Even after applying the hardware-embedded low-pass filtering on the sensor itself, it still incorporates noise due to the aliasing effect. A threshold above the noise level is chosen to derive the true signal component from the underlying noise components. If it is set too high, less signal will be registered and thus made the EPS-based smart floor insensitive to weak signals especially in case of movement without footwear. Therefore we conducted the experiment of first varying the threshold for the test runs without foot wear. This gives us the upper bound, where this threshold should lie within. We discretized each measured floating point sensor value to a 12-bit integer value. Two thresholds of 50 and 100 were tested to illustrate the performance of localization accuracy. The mean positioning error increased from 16.34 cm for the threshold of 50 to 19.7 cm for the threshold of 100. Thus we chose a common threshold of 75. In accordance to the selected signal threshold, we determine whether an electrode is active or not. We denote a sensor reading of a horizontally oriented electrode wire h_i as r_{h_i}. Likewise, we denoted the sensor reading of a vertically oriented electrode wire v_j as r_{v_j}. Per floorspace unit, we then have certain electrodes of interest forming floorspace activities $FA = (h_i, r_{h_i}), ..., (v_j, r_{v_j}), i = 1..N, j = 1..M$, where N and M are IDs of electrodes according their spatial placement. As can be seen from Figure 6.13, the impact area is correlated to the underlying signal strength. The strongest signal is right beneath the point of impact from the foot motion and weakens towards the outside. How strong this induced charge is, depends proportionally on the strength of the inherent body electric charge.

Region of Interests per Floorspace For each 100 ms a snapshot of sensor values is taken. A region of interest (*ROI*) is formed out of a subset of floorspace activities (*FA*) which belong together. For each *ROI* certain activated horizontal and vertical sensing electrodes close by will be combined to form sub-regions of activities within one floorspace unit. We use $ROI = (h_i, r_{h_i}), ..., (v_j, r_{v_j}), i = a..n, j = b..m$, where n and m were the IDs of respective sensing electrodes, to illustrate one of such sub-region area. Basically, one illustration of such a ROI is depicted in Figure 6.13 by the colored electrode wires underneath the footstep. The range of impact could vary due to the body electric charge. Due to multiple events per floorspace, it is also possible that one can find more than one

region of interest. In order to realize multiple person tracking, it is important to allow the existence of multiple region of interests $ROIs, s = 1..K$ within one floorspace. In our proposed algorithm, we also include activated antenna which is within one inactive antenna apart still be clustered into the same region of interest. Thus, an activated antenna which is more than 40 cm away from an inactive one is clustered into a new region. This makes the system to resolve two persons, if they are standing more than 40 cm apart from each other. Otherwise, they could be merged to one large area of interest.

Center of Activities Now we calculate the center of activity for each ROI_s containing n activated horizontal electrodes with (h_i, r_{h_i}) and m activated horizontal electrodes with (v_j, r_{v_j}) using a weighted average method by applying the Equation (6.6):

$$\begin{aligned} x_{ROI_s} &= \frac{\sum_{j=b}^{m} r_{vj} x_j}{\sum_{i=b}^{m} r_{vj}} \\ y_{ROI_s} &= \frac{\sum_{i=a}^{n} r_{hi} y_i}{\sum_{i=a}^{n} r_{hi}} \\ A_{ROI_s} &= \frac{1}{m}\sum_{j=b}^{m} r_{vj} + \frac{1}{n}\sum_{i=a}^{n} r_{hi} \end{aligned} \tag{6.6}$$

where the positions of x_{ROI_s} can be extracted using the activated vertical electrode and the positions of y_{ROI_s} can be extracted using the horizontal electrodes. The third component A_{ROI_s} provided the strength of this center of activity. The strength is directly proportional to the probability of the presence of the person in that location which kind of represents a degree of belief. With increasing A_{ROI_s} the probability increases that the person is localized on this position (x_{ROI_s}, y_{ROI_s}). This way, we calculated all the center positions of all possible K regions of interest $ROIs$, $s = 1..K$ within one floorspace with its own degree of belief.

In the current work, we only investigated and evaluated the single user tracking, due to limited experimenting area. This proposed concept could be extended to multiple person tracking. To adapt the current algorithm to perform multiple person tracking, we should pay further attentions to the found region of interests and not only select the one with the most probable center position and discard the other positions. A concept of multiple person tracking is provided in the later section.

Tracking of Center Positions In this work, we implemented a basic mean shift tracking algorithm to build the motion model. Relative to the possible options of the sub-regions of activities ROI_s, $s = 1..K$ and their center of activity (x_{ROI_s}, y_{ROI_s}), where $s = 1..K$, the most probable location needs to be chosen. Upon the first detected location, the position with the strongest belief A_{ROI_s} is chosen to indicate the most probable location and the successive positions are always chosen with respect to the last found position according to A_{ROI_s} from current step.

Multiperson Tracking Here, we describe conceptually how our existing algorithm can be extended to perform multiple person tracking and how we try to overcome the problem of occlusion.

- A person will be created if there is a valid position measurement more than the current number of existing person and could therefore not be assigned to any of the existing person. Creating a new person is only valid if the distance to an existing person is large enough. The minimum distance is at least 40 cm in our proposed application.

Figure 6.14.: Test setup in the living lab with electrode grid and sensors.

- A person will be updated according to the assignment rules used in Hungarian algorithm as proposed by [KY55]. Kalman filter is used to filter out unsatisfying tracking results due to different real-time conditions including inter-object occlusion, splits and merges, which can be observed when targets are being tracked in real-time.
- A person will be deleted if she/he can not be updated for a longer period of time or if the person leaves the room through one of our predefined exiting areas such as doors or balcony doors.

6.2.2.3. Experiment and Evaluation

In order to evaluate the spatial accuracy of our indoor localization algorithm we designed a study. We were especially interested in investigating various environmental influences on the EPS-based smart floor, such as different footwear like the sole materials and the effects of walking barefoot. From theory, we know the charge accumulation is less prevalent when walking barefoot. Thus, the voltage change induced on the measuring electrode is much weaker. This scenario represents our worst-case scenario.

For the test study, we asked 12 participants to walk on a predefined path. The group of subjects including two females and ten males, between 23 and 37 years of age (average age 27), with a mean weight of 88.8 kg for the male group and 53.5 kg for the female group. We do not separate the impact of body weight and footwear. All impacts result into sensor readings. Each test run followed a given path, which has been previously marked on the ground. The markers were set 50 cm apart simulating a stride length of an elderly. For each participant four test runs were expected, consisting of two test runs wearing shoes and two test runs with bare feet. The evaluation took place in our living lab, where we installed our indoor positioning system using two subsystems each containing ten measuring sensors, as depicted in Figure 6.14.

We selected a walking speed of one step per second to simulate a slow human walking speed. Each second, a timer played a tone to help participants adjust their walking speed and to trigger the processing. We compare

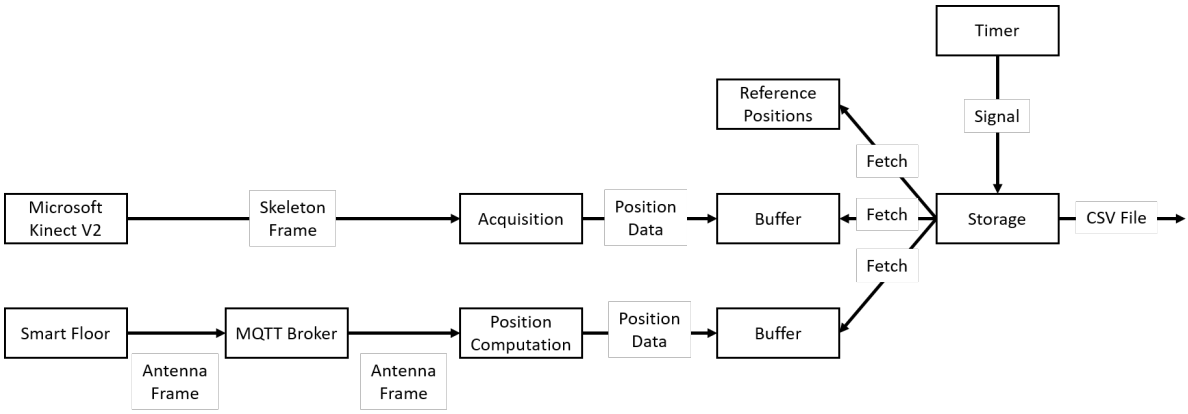
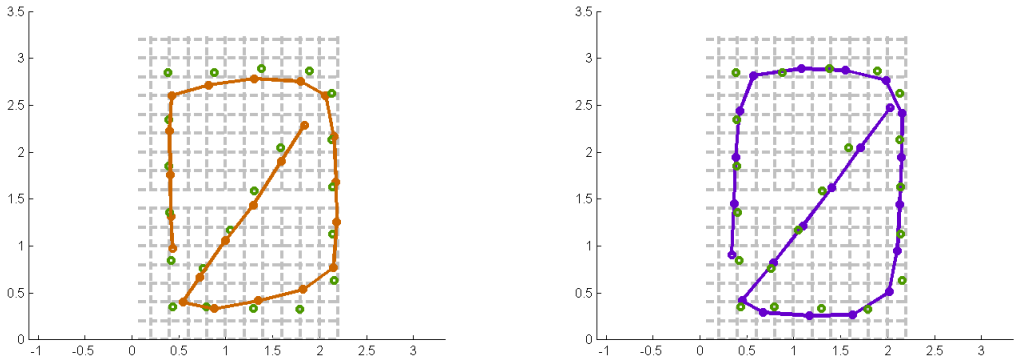


Figure 6.15.: The building components of our evaluation system is depicted here. We collect position information from Microsoft Kinect V2 and the EPS-based smart floor to compare them to reference positions.

the performance of the EPS-based smart floor and a Microsoft Kinect V2. For this task, we mounted a Kinect horizontally on the side of our experimental setup to use the skeleton positions to track the person's movement. Since the entire area can be fully covered without occlusion, this placement seems to be reasonable. Other possible setups like ceiling-mounted Kinect is also possible. However, since a skeleton tracking without further effort is not possible for ceiling-mounted Kinect, an additional algorithm to track the position of the head is required. Therefore we discarded the idea of ceiling-mounted Kinect for the sake of simplicity. For each marker, we determine the x- and y-coordinates to the origin point of the EPS-based smart floor evaluated from the set of collected sensor values of the system. The data acquisition of the EPS-based smart floor with the evaluation program communicates over an external MQTT server. With every new message, the position was determined and stored in the buffer. The Kinect acquisition used the connected Microsoft Kinect V2 device and Microsoft Kinect API to retrieve the skeleton position to compare to the EPS-based smart floor. Both position data will be first stored in the buffer and fetched shortly after the timer is activated. Upon each timer activation, we fetch and store all position data with the reference position in different stored text files for further processing and evaluation in Matlab R2013b. A system overview with the building components of our evaluation system can be found in Figure 6.15.

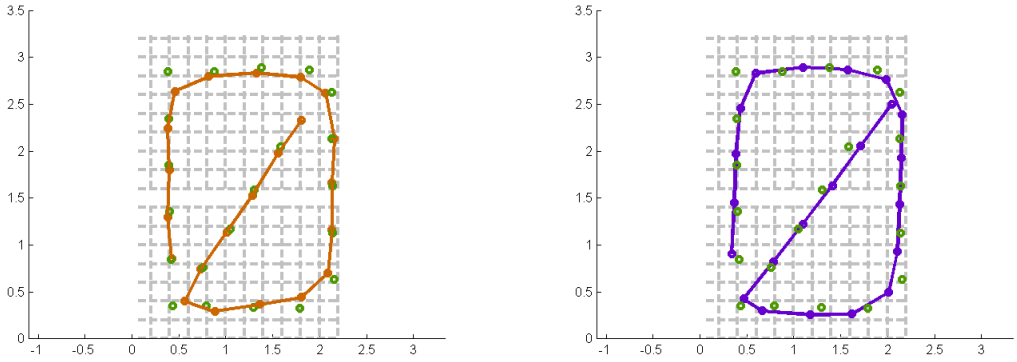
To determine the impact of various footwear to the performance of the EPS-based smart floor, we divided our test runs into two different sets. One set included all test runs conducted with participants wearing shoes and one set contained all the remaining test runs conducted with participants walking barefoot on the sensing area. All conducted test runs with their resulting trajectories can be seen in Figure 6.18 and Figure 6.19. In Figure 6.16 and Figure 6.17, the average mean path for all test runs is depicted for each setting. This gives us an impression of the average position errors with respect to the reference positions illustrated as separated dots. In Figure 6.16(a), the mean trajectory without shoes is depicted while in Figure 6.17(a), the mean trajectory is shown when the participants wore shoes. First visual inspection indicates only slight difference in the mean positioning error for both experimental settings. To grasp the fine difference, further statistical tests are required.

Obviously for Kinect-based system, the accuracy of the trajectories are independent of external influences such as footwear. This is illustrated in Figure 6.16(b) and Figure 6.17(b). In order to show the significance of footwear towards the EPS-based smart floor, we further conducted the non-parametric Kruskal-Wallis H test as introduced by [KW52] on our collected data points. To quantify the accuracy of the EPS-based smart floor, we denoted the a priori known real-world positions of the reference path as p_{ref_i} in the form of $p_{ref_i} = (x_{ref_i}, y_{ref_i}), i = 1, ..., n$



(a) Mean Trajectory without Footwear Proposed (b) Mean Trajectory without Footwear Kinect

Figure 6.16.: The orange curve shows the mean path on each position averaged over all the test runs with test persons walking barefoot. The purple curve shows the mean path from the comparing system using Kinect 2.0. The green dots represent the marked reference position on the floor.



(a) Mean Trajectory with Footwear Proposed (b) Mean Trajectory with Footwear Kinect

Figure 6.17.: (a) The orange curve shows the mean path on each position averaged over all the test runs with test persons walking with shoes. (b) The purple curve shows the mean path from the comparing system using Kinect 2.0. The green dots represent the marked reference position on the floor.

setup	mean	median	standard deviation
with shoes	12.7cm	7.4cm	13.6cm
without shoes	18.0cm	8.8cm	22.1cm
Kinect Hip	15.4cm	14.7cm	7.4cm
Kinect Right Foot	22.0cm	15.2cm	15.8cm
Kinect Left Foot	20.7cm	13.9cm	16.1cm

Table 6.7.: Positioning errors compared to pre-marked reference positions over all test runs with individual settings.

and the test path as $p_{test_i} = (x_{test_i}, y_{test_i}), i = 1, ..., n$. The distance error was calculated using the Euclidean distance $d_{err} = ||p_{ref_i} - p_{test_i}||_2$ on each prior marked reference position. For each test run, we collected 22 distance errors between test position p_{ref_i} with respect to reference position p_{test_i}. Based on the data collected from 12 test persons, each walking two times in each setting, we recorded 22 x 2 x 12 = 528 distance errors for recordings with shoes and additional 528 distance errors without shoes. The results show that walking barefoot had significantly different outcomes compared to walking in shoes ($p < 0.001$). The overall mean positioning error was 18 cm, with a standard deviation of 22.05 cm for recordings without shoes and a mean positioning error of 12.7 cm with a standard deviation of 13.6 cm for recordings with shoes, as shown in Table 6.7. The error deviation in case of wearing shoes is significantly lower compared to the case of walking bare foot. This inferior performance of the EPS-based smart floor from wearing shoes compared to without can be explained by the nature of our measuring principle itself. While the person walks barefoot on the sensing area, the charge separation induced by human walking is too small and, thus, drains off too quickly. On the contrary, for a person wearing shoes, the isolating sole material keeps the human body charge separated via walking constantly when it induces a change to the electrode.

All trajectories for all participants are depicted in Figure 6.18 and Figure 6.19. We draw the positions from the EPS-based smart floor compared to the reference positions to illustrate the overall performance of our proposed smart floor. Peculiar are the first two runs from participants 1 (c-d) and 2 (c-d) without wearing shoes. After interviewing the participants, this could be explained by the reported changed habit or uncertainty from walking in shoes to without shoes. Given the instability of walking barefoot on the marked positions, these participants felt it was difficult to walk naturally on the given path.

6.2.2.4. Discussion and Conclusion

Walking is a principal movement for humans. The biped motion is generally divided into a single support phase when only one foot is on the ground, and a double support phase, when both feet are on the ground as explained by [RZH05]. Stride and step length are dependent on various factors, such as body size, the position of the feet, or hip mobility. These vary from person to person and even change with age. According to [ETHC91], compared to young adults, the elderly exhibit 17 to 20% reductions in the velocity of gait and length of stride. Therefore, in order to keep the computational effort for pure tracking low, a maximum stride length of 80 cm was assumed, according to [IKH13]. With a operating frequency of 10 Hz, our proposed system is able to resolve a step width around 8 cm per sampling unit. According to the localization accuracy of our proposed system, we conclude that our system is able to perform accurate indoor tracking although with a moderate system operating frequency.

In view of the thorough study of our proposed system, we further demonstrated that the EPS-based smart floor possesses certain preferable advantages compared to vision-based systems such as a Microsoft Kinect. The EPS-based smart floor tracks the person in the entire room without the problem of privacy invasion and occlusion. The

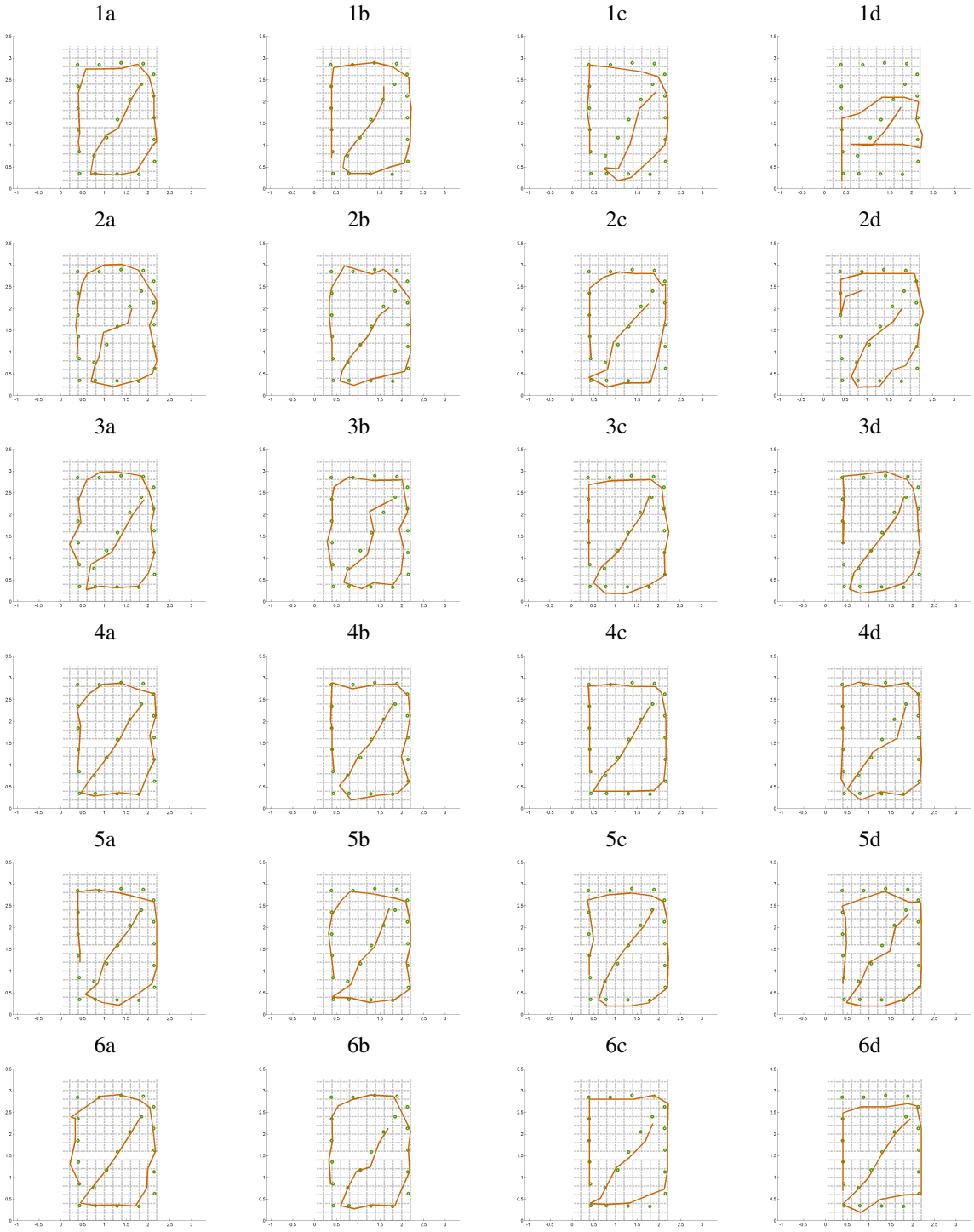


Figure 6.18.: EPS trajectories for participants 1 to 6 is illustrated, while rows (a,b) are runs with shoes and rows (c,d) are runs bare foot. Green dot indicates reference position.

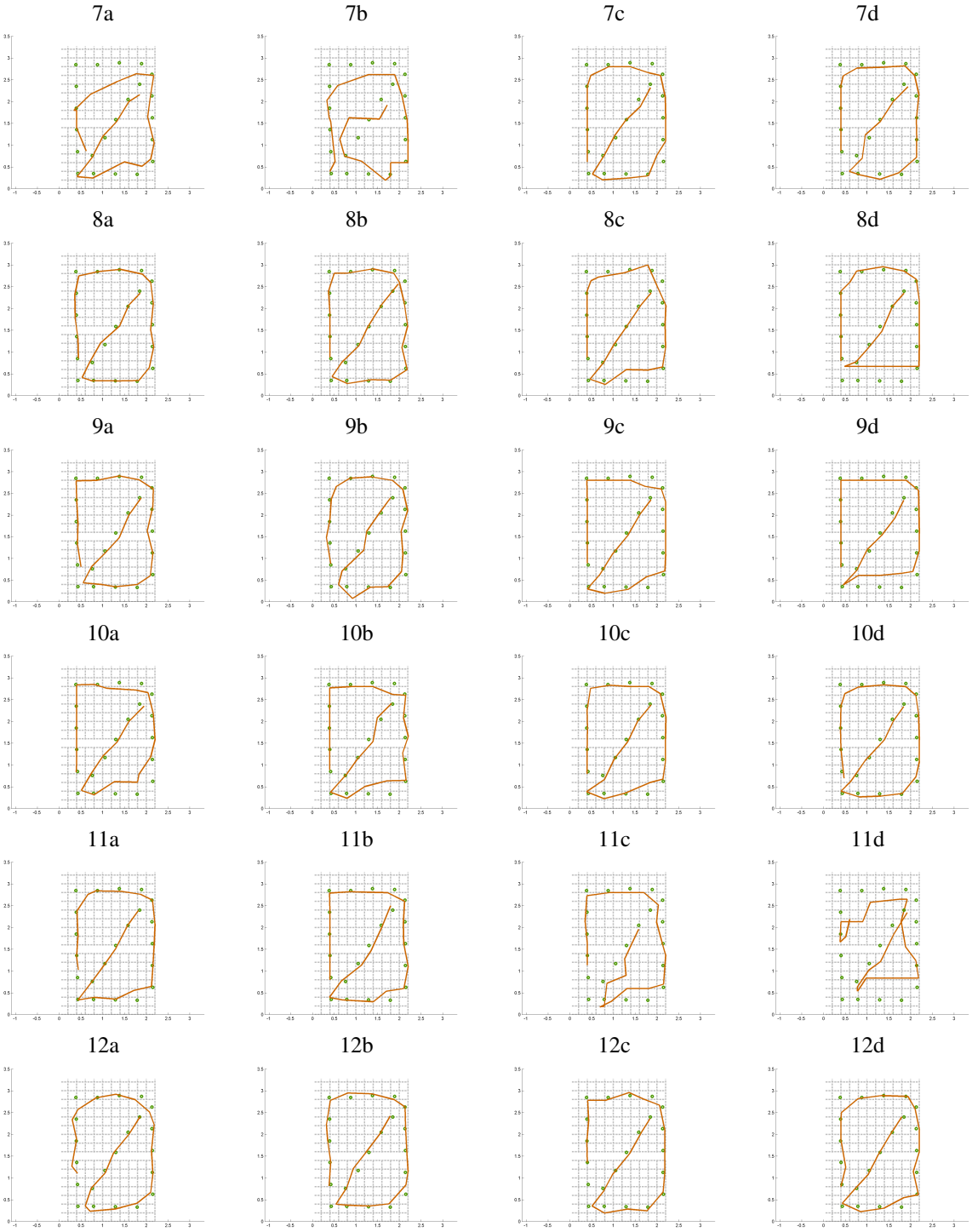


Figure 6.19.: EPS trajectories for participants 7 to 12 is illustrated, while rows (a,b) are runs with shoes and rows (c,d) are runs bare foot. Green dot indicates reference position.

feature	−−	−	*o*	+	++
res	very coarse	coarse	normal	fine	very fine
upd	<1 Hz	<10 Hz	25 Hz	>50 Hz	>100 Hz
det	touch	<1 m	<5 m	<20 m	>20 m
unob	open large system	open small system	hidden, large exposure	hidden, noticeable exposure	invisible
proc	single sensor CPU	10+ sensors CPU	single sensor, embedded chip	10+ sensors by single chip	no further processing
calco	very hard	hard	normal	easy	very easy

Table 6.8.: Feature matrix denoting capabilities required for a certain rating. List of Features are Resolution (res), Update Rate (upd), Detection Range(det), Unobtrusiveness (unob), Processing Complexity(proc), Calibration Complexity (calco).

System	res	upd	det	unob	proc	calco
Camera	++	o	+	-	o	-
WiFi	-	+	++	+	o	o
Depth Camera	+	o	o	-	-	o
PIR	-	+	o	-	+	o
Capacitive	-	+	-	++	o	+
EPS	+	-	o	++	+	+

Table 6.9.: Here token-free systems commonly used for indoor localization are compared to each other. The voting are based on the features introduced in Tab. 6.8.

most important advantage is its low energy consumption and accurate localization due to its passive measuring nature.

In Table 6.9 we benchmarked our proposed system compared to other common token-free indoor localization systems proposed in the related work section. The assessment is related to the feature matrix provided in Table 6.8. The different systems show their own challenges and limitations. The EPS-based smart floor, however, outperforms most of the listed common indoor localization system concerning its low processing complexity, low energy consumption, and easy maintenance. Therefore, such a smart-floor system is suitable for various smart home applications. It can be used e.g. in elderly care to detect falls in the apartment. Caretakers or nursing stuff can be automatically informed after a fall is detected and an alarm is set. The signature of a fall is distinctive to the signature of a simple step signal as depicted in Figure 6.20 and can thus be detected.

The system can be used in anomaly detection of daily activities. A database can be used to store all trajectories taken within days or months. According to the recorded behavior patterns and habits, behaviors that deviate from the norm can be identified and reported if necessary. It can be used not only to perform behavioural analysis, but also to be used as an early warning system for the early detection of mental disabilities or deficiencies.

The system can be used in burglary detection. The residents can set a time of absence when they are on leave. By detecting unauthorized activities within this predefined period of time, an alarm can be triggered to inform the person in charge. If certain activities close to windows or doors are detected, the police can be notified if the alarm is not deactivated within a certain period of time.

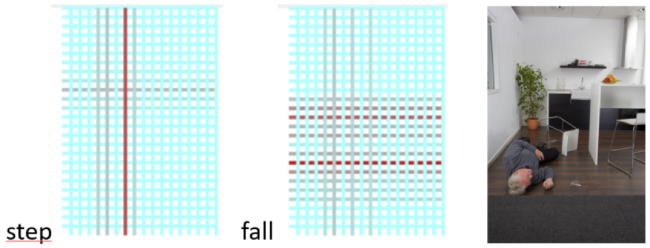


Figure 6.20.: Here depicted are an momentary snapshot of a step signal and a fall signal. As can be observed, the fall signal is clearly different and dominant than a simple step signal.

In this Section, we have shown that by leveraging a floor-based system with electrostatic sensing for indoor localization, even with a moderate operating frequency of 10 Hz, we can still achieve a sufficient precision in localization accuracy. Scherf [SKvW*18] proposed a way to perform early detection of dementia or other mental disabilities from elderly, with our proposed hardware infrastructure and the indoor localization algorithm in a real apartment setup. Thus, the current system is able to perform precise indoor positioning and recognition of abnormalities based on trajectory tracking build upon our localization algorithm.

6.3. Summary

In this chapter, I proposed two floor-based stationary applications used in the smart home context with the focus of addressing the main research question *how to setup a successful HAR system*. I addressed this issue by applying the previous findings from the main components within the proposed framework covering its individual challenges starting from sensor selection, data acquisition, data processing, modelling, and real-world deployment.

One application is based on the surface acoustic events measuring force-based events, while the other is based on electric potential changes caused by varying body potential charge via movement. Both are suitable sensing technologies for constructing appliances for a smart environment at different scales. Surface acoustic sensors can differentiate more fine-grained vibration patterns compared to electrostatic measurements within a restricted area with good vibration transmission property. Though it is more restricted to constraint environments due to its coupling issue and the material-dependent transmission and attenuation properties. Here, I would shortly recap both system designs relate to the basic components in the overall design framework proposed in Figure 1.1.

6.3.1. Surface acoustic sensors

Four pickup sensors are used to form the array covering an sensing area of the dimension 3.9x3.42 m covered with a parquet floor to enable a better surface wave transmission. The preliminary task for this system is to recognize a set of activities of daily living. The difficulties in this task resides in the similarity of certain subtasks, such as considering step signals from bare foot, with shoes or high-heels signal. Differentiating objects interaction events from different positions is also challenging. Further investigation of minimalist sensor setups is performed to test the viability of applying less sensors without losing classification performance.

Sensor selection As applications designed in this chapter are for home environments only, it poses restriction on possible sensor categories. Camera-based application is often banned due to privacy issue. Complex systems are not suitable for an easy integration into existing infrastructure. Therefore, I voted to use commercial hardware, such as a M-Track Audio device including pick-up sensors with minimalist requisition on external hardware design.

Data acquisition For this phase, an external storage is required, such as a laptop to store the impact-based vibration patterns. An audio recorder script is written with PyAudio package to record and a wave library for storing the wave files is used. The participants were asked to record and label the activities on their own. The sampling frequency of the M-Track audio device is device-specific and is set to 96 kHz. Channel synchronization is performed to reduce the systematic errors caused by the system setup. With regard to improve the signal-to-noise ratio, a sound coupling between the pickup and the ground surface is essential.

Data processing Per channel normalization is applied to reduce the hardware specific channel effects. Two different window sizes are investigated to evaluate the system performance. Hand-designed feature extraction is performed in time and frequency domain. In the time domain, the time difference of arrival, the root mean square of signal segment and the zero crossings are calculated. Frequency domain considers the Fourier components and other frequency-related properties.

Modelling Due to the low dimensionality of the hand-designed features and the discriminative nature of the samples, I applied conventional machine learning methods, such as support vector machine, decision tree and the naïve Bayesian network to classify the segments. This step is kept modular, as different classifiers can be implemented and exchanged on the feature set.

Real-world data By collecting signal patterns from different scenarios for a step signal, such as walking bare foot, walking with shoes, or walking with high heels, I consider more realistic real-world data. Here, I distinguish the fine-grained differences in a step signal for a real-world application.

Collecting cupboard events on different locations further indicates the usefulness of including directional or location information of the origin of the sound impact. The three cupboard doors have the same material and size properties. The only difference is the position of the cupboard in the room.

Online application Compared to other vibration patterns, the fall has more discriminative features as it contains higher frequency components. Precise fall detection results for the proposed application indicate the usefulness of such a system for a real-time fall detection for elderly in a smart environment.

6.3.2. Electrical potential sensors

In Section 6.2.2, I proposed a floor-based positioning system with electrostatic sensors. A grid-based layout is used to scale the system towards different room geometry. I underline the ability of the proposed system to provide indoor positioning data to be further reasoned to build more complex contexts and thus enabling more complex appliances or services used for smart home applications.

Sensor selection Regarding this application, I have the luxury of freely designing the system with customized hardware and software setups. As the proposed system should be carefully planed and installed during the construction phase of the building, it is already integrated into the infrastructure after construction. Considering the advantages of easy maintenance, low power consumption, large detection range and higher sensitivity, I decided to use electrostatic sensors in the proposed system.

Data acquisition The system operates at 10 Hz. According to the research of human locomotion on step signals [SKBM*97], I assume an average gait velocity of $1.5\frac{m}{s}$ and an average step width of 81 cm. The system resolution in time and transmission load make 10Ḣz sufficient to sample the gait action. Data storage is available directly on the Beagle Bone Black (BBB) as the central processing unit. For evaluation purposes, I have performed the processing on the BBB and forwarded the calculated system positions to be recorded on a desktop PC.

Data processing Electrostatic sensors are susceptible to environment noise, power lines, or other electric appliances in the vicinity. These noise sources are easily coupled to the sensor. Even with a hardware filter, these effects are not negligible. In favor of reducing these negative effects, a signal threshold is introduced to filter out these noise using software filters. The exact calculation of the indoor position is directly correlated to the strength of the activated sensor values.

Data Modelling Current position is calculated using the weighted average methods of the activated sensor grids. A basic mean shift tracking algorithm is used to smooth the positioning information. No further learning is required. One disadvantage of this sensing technology is that it only measures dynamic signals. Thus no position updates is possible, if the person is stationary. Rule-based handling of this effect is therefore critical in order to perform a stable person tracking.

Real-world data Towards simulating the real-world data, I collect step signals both from the trial runs with and without footwear. I demonstrated the impact of different footwear in relation to the strength of the induced signal and the accuracy of the indoor positions. Without considering these effects and impacts, the system would under-fit to real-world data by underestimating the real effect.

Online application All processing is performed locally on the BBB, information stay within the apartment to preserve privacy. As the BBB only has limited storage and processing capability, only limited statistics such as the current status relate to the presence or absence of the inhabitant and position information of the last active path are stored to save memory.

7. Conclusion and future work

Derived from the main research question in this thesis with respect to build successful sensor-driven application for HAR, I proposed the common framework as depicted in Figure 7.1. This structure has proven its worth throughout my work experience with these sensor applications. Related to this framework, I further identified connected challenges within each main component along the pipeline. These challenges further result in the subsequent 6 research questions this thesis is dealing with. A detailed response towards the research questions can be found in the previous chapters (2, 3, 4, and 5).

In this chapter, I first summarize my contributions with respect to the posed research questions in Chapter 1 in Section 7.1. Finally, I conclude this chapter while providing interesting future research directions as guidelines for myself or other HAR practitioners in the field of designing sensor-driven applications for HAR in Section 7.2.

7.1. Conclusion

The goal of this thesis is to develop sensor-driven applications for human activity recognition (HAR) in smart environment beyond the common sensor technologies currently used. HAR is a wide research field containing several subfields, such as physiological sensing, indoor localization, gesture detection and behavioural analysis among others. Great amount of research works have been done in various subfields of HAR. I proposed a framework in Chapter 1 in regard to the best practice knowledge I have gathered during my research within this field. This framework aims at solving the main research question of *How to setup a successful HAR tool chain*. This framework contains the topics of sensor selection, data acquisition, data processing, data modelling and real-world deployment. It is set out to assist designing sensor-driven applications. The main framework with its individual challenges finally result in the subsequent research questions this thesis dealt with.

Research question 1 *Which **sensor category** has to be applied under which conditions?*

To address this research question and to identify possible scientific contributions to still unexplored domains in the research field of HAR, I conducted the survey in Chapter 2. Based on my proposed sensor categorization scheme according to the physical entity they measure, it allows me to draw comparisons across sensor categories for certain application areas. I further revised the most prominent works utilizing these sensor categories in the domain of HAR. The use of sensors in smart environments has increased with the decreasing manufacturing price for miniaturized sensors. The demand on the skyrocketing number of IoT devices in smart environments makes sensor-driven application for HAR a very import research direction. I identified research gaps within possible applications in Quantified-self domain using sensor categories that are more robust and flexible compared to existing sensing technologies in this domain.

The demand on Quantified-self applications has been increasing in recent years, with numerous miniaturized sensors integrated in smartwatches, smartphones, or other smart devices. In this area, camera-based or body-worn systems are predominant. While they achieve high accuracy, these methods often suffer from privacy issues or obtrusiveness and consequently social stigma. In Chapter 3, I proposed two different sensing technologies with

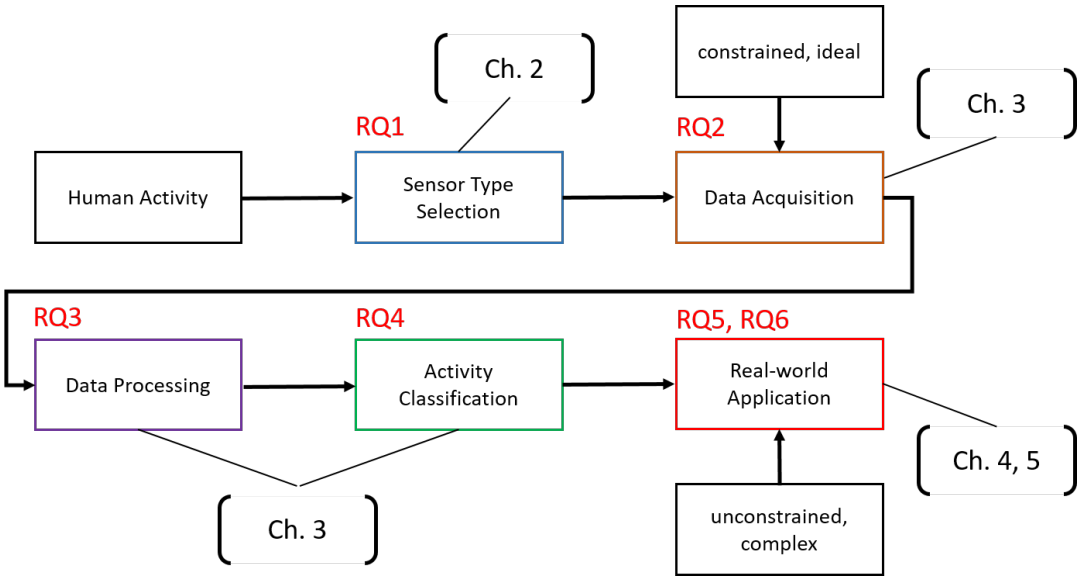


Figure 7.1.: Illustration of the generalized framework to successfully setup a HAR system. The main components and their individual challenges result in the subsequent research questions of this thesis.

the remote sensing capability to detect exercises without the limitation of wearable designs or visual camera inputs.

The first application in Section 3.1.3 leverages ultrasound Doppler sensing characteristic with a commercial smartphone to recognize eight whole-body exercises. A person's smartphone provides a richness of information. The huge variety of sensors in today's smartphones makes these devices an outstanding target platform for human activity recognition. The fact that phones nowadays come equipped with a variety of sensors, makes them an excellent platform to infer the context and activities of the respective user [Sch00]. I successively construct the application by first performing a thorough experimental study. Based upon the conducted exploratory study, I illustrate the feasibility of a smartphone using its native microphone and speaker to recognize nearby movements. I then further extended the application to recognize more complex and realistic exercises in the successive sections.

The second application in Section 3.2.3 constructs a mobile prototype with capacitive proximity sensors embedded in a consumer yoga mat. This application aims to improve a similar pressure-based textile application for physical exercise recognition with improved robustness against system deformation. In contrast to pressure-based sensing, capacitive proximity sensors enhance the sensing modality by sensing objects up to 15 cm distance without enforcing a touch interaction. This provided us the possibility of detecting more fine-grained activities in mid-air. The symmetrical sensor placement further allows me to perform data augmentation methods to increase the model generalization ability for real-world applicability.

Related to the knowledge gathered from designing these prototypes and evaluating their performance and functionality, I am able to answer the previously posed research questions (2), (3) and (4) related to these specific sensor-applications.

Research question 2 *What has to be considered for **data acquisition** with specific sensor technology?*

Addressing this research question, several basic design choices in preparation for the data acquisition phase should be answered. These design choices are such as the investigation of the sampling frequency with respect to hardware specification and task specific activity velocity. Sensor placement is explored by applying structured tests to optimize the signal-to-noise ratio. In addition, the environmental noise coupling characteristic to the sensing technology is verified to mitigate noise interference to the application.

Sensor application with freely adjustable sampling frequency should set this parameter according to the task. For instance, according to the speed of human locomotion in the literature, the sampling frequency should be at least 20 Hz (as derived from Section 3.2.3.1) to target the fastest speed of sport exercises used in our designed application with the capacitive sensing yoga mat. If the sampling frequency is set too low, the resolution in time is not enough to resolve the fastest human motion. Sensor placement, its shape and size have a strong impact on the signal strength and thus can modulate the signal appearance. Ultrasonic sensing for example relies strongly on the sensor placement due to the material-dependent attenuation effect of the physical entity. Placing the mobile device close to a wall corner, the wall behaves like a corner reflector and increases the back reflected signal strength. This behaviour is derived from experiments conducted in Section 3.1.6.1. In addition, the electrode shape and size form the sensing field and thus determine the detection range, signal-to-noise ratio and the signal resolution.

Last but not least, the labeling process is one of the most tedious step in building sensor-driven applications for HAR. The labeling process is error prone and the most time consuming. Solving this issue, I used two separate approaches. In the first design, the user has to insert the labels on their own. The application is designed such, that the user has to select the exercise before performing them. The intention of using this approach is to pose as less intervention as possible to the user. In the second design, an instructor is assigned to label the data, while the participant is performing the activity. In this way, the quality of the labeling process is consistent. However, since the targeted activities in this thesis are well defined, the labeling process in both approaches are straightforward and comparable.

In conclusion, there exist certain basic pre-design choices related to the application which should be taken into consideration in the data acquisition phase. A short summary of this basic design steps are provided in the summary of Chapter 3.

Research question 3 *What degree of **data processing** is sufficient without affecting the performance of the data modelling?*

Time series segmentation into appropriate window length is researched according to model performance for offline processing. This step is called windowing. In this thesis, I investigated the performance of the model accuracy against window length. A too short time window does not fully cover the entire activity of interest within the segment and thus leads to a poor performance. Selecting a time window wider than necessary also leads to noise inclusion. Therefore, the appropriate length of time window segmentation is according to the duration of the activities of interest.

I further compared the conventional handcrafted features from time and frequency domain to methods using automatic feature extraction as a built-in component in the model. Handcrafted features such as means, variances, further statistical moments, zero-crossings, Fourier frequency coefficients are calculated from the window segments. Dimensionality reduction technique was applied on these engineered features to reduce the internal correlation within the data and reduces the successive computation effort. The performance of building models using these conventional features are compared to 2D-CNN models where the low-level feature extraction is integrated into the network itself.

To give a final remark on the research question (3), we have to be aware of the applied model architecture. Therefore, this question can not be answered without simultaneously considering the research question (4) in the next paragraph.

Research question 4 *Which architecture or model has to be used for **activity classification** in certain sensor application of HAR?*

This research question is answered by carefully reviewing specific designed models used for individual tasks. The amount of data imposes limitation on the model capacity. For small amount of data, conventional machine learning model can outperform the deep learning models. However, with the increasing amount of data, the performance of conventional machine learning approaches will saturate at some point, while deep learning techniques clearly benefit from the increasing amount of data.

Multidimensional time series, such as acceleration data, can be perfectly modeled by using sequence modeling architectures. Two dimensional data that resembles image data, such as time series signal transformed into time-frequency spectrum, can be modeled by leveraging two dimensional feature extraction filters. Discriminative methods, such as SVM and logistic regression perform well on separable handcrafted features. The feature engineering relies however strongly on inductive biases, such as including domain knowledge of a particular field. The end-to-end learning classifiers combine feature extraction and classification in one optimization step.

In Section 3.2.3, conventional models such as k-NN and SVM show good performance on separating user-dependent exercise data. However, k-NN appears to have the most trouble with the *None* class separation and with some exercises performed in the lying position. Highly nonlinear models such as the parallel branches late fusion (PBLF), which consists of mostly CNN features, achieve better recognition rates for all exercises. There is no clear indication on which model performs the best. It strongly depends on the underlying data structure. For high-dimensional nonlinear feature distribution, the machine learning practitioners may benefit from the complex model capacity of the end-to-end learning models. End-to-end learning shows good property in separating strongly nonlinear high dimensional data. Low-dimensional features or features with good separability can be well classified by using conventional machine learning models.

Up to this point, I have been mainly working on functional prototypes for certain sensor applications. The following two research questions target at how to deploy the designed prototype to fit to real-world scenarios and how to improve the model generalization ability on real-world data.

Research question 5 *How to overcome the gap between constrained development data and the more complex **real-world data** with the scope on time series for HAR applications?*

While addressing this research question regarding the real-world applicability of our developed applications, two contributions were made on the adaptation of real-world data. We approached this question both from the data space and the feature embedding space. In Section 4.1, I approached the research question by increasing the variability in the data space. This selected approach was validated on the developed application with the enhanced yoga mat in section 3.2.3. The objective was to show that data augmentation on time series is needed to train more generalized inference models. I compared conventional against generative methods. Provided by the symmetrical setup of the prototype, conventional data augmentation was realized by simply flipping the order of the time series. More complex methods, such as time or amplitude warping, generated more diverse time series data. Generative methods were used to approximate the hidden representation of the input data distribution and therefore generate synthetically fake samples realistic enough to mimic the original data drawn from the same source distribution. Therefore, by adding samples generated from the generative models, I could further increase the diversity in data in a supervised manner. These proposed methods succeeded in improving the model performance on unseen data by properly diversifying the training data and decreased the model variance

on unseen data. However this method comes with the restriction that the development distribution is similar to the real-world data distribution. Large difference in both distributions makes this technique inappropriate to generalize the inference model.

In Section 4.2, I addressed the research question of increasing model generalization ability for the mobile applications developed in Section 3.1.7 targeting finetuning on individual fitness tracking. Here, I face the gap between the data collected and trained under controlled environment versus the uncontrolled data collection in the real-world scenario. To minimize the gap, I aim at solving this problem in a metric-based approach from the feature embedding space. By using the base trained model with data collected under a controlled setup, I intend to increase the adaptability of data collected under more realistic conditions. The retraining with only limited amount of new data leads to model overfitting. To overcome this problem, I showed that individual finetuning with few shot learning techniques are most suitable resolving this issue and provide improved generalization ability with only few data samples. By incorporating only few labeled data from the new setup, I am able to achieve good classification results and improve the model generalization on the new data without retraining the base inference model. Similar model improvement can also be achieved by using domain adaption techniques. Domain adaptation technique can be used to close the feature gap in the embedding space between the controlled data domain and the uncontrolled data domain. However, this method requires the model to be retrained by simultaneously learning common features from both data domains. This enables a good classification both in the controlled source domain and also in the uncontrolled target domain, assuming both domains have certain base similarity.

Research question 6 *How to scale model complexity used in **real-time applications**?*

Chapter 5 is dedicated to deal with this research question. Possible solutions to this research question is demonstrated by using two applications proposed in this chapter. I first describe how to deploy application in Section 3.2.3 with the enhanced yoga mat to run on a Raspberry Pi 3 with restricted processing capability. It was realized by implementing end-to-end learning network to mitigate the stage of handcrafted feature calculation. Fixed inference time due to fixed model of the learnt network is necessary requirement to build real-time systems. I then introduce in Section 5.2 a model-based approach for the capacitive mid-air gesture recognition platform to be implemented on a standalone device. This is realized by restricting the feature dimensions and working with a simplified model-based approach. The tag-free indoor localization system in Section 6.2.2 with electrostatic measurement is also realized for running the positioning algorithm on a Beagle Bone Black (BBB) processing unit. As the BBB only has limited storage and processing capability, limited processing results are stored on the device itself. However, this approach preserves privacy while keeping the motion histories inside the apartment.

The previously discussed solutions contribute to different components of the proposed framework provided in Chapter 1. Here, I only summarized the joint conclusions from the experiments conducted in this thesis. The details towards the individual contributions can be found in previous chapters.

7.2. Future work

Grounded on my current research results in this thesis, I identified the following aspects which seem to be interesting and worth investigating in the future.

Enhance location context In Section 6.2.2, I have presented a floor-based indoor localization system with electric potential sensors. Considered as one type of sensor, the location information offers a basic context which can be integrated to form a broader context by using a reasoning system. Future research direction in healthcare

domain is to integrate the base location context to recognize patients with dementia or pain detection for elderly inhabitants. In order to build entertainment applications in smart home, I can imagine to construct personalized settings based on identification of gait or trajectory patterns.

Multi-modal sensor fusion In this thesis, I have been working on developing single sensor-driven applications to target at certain sub-fields of HAR. Each sensor technology provides special types of context information. A promising opportunity is to fuse multiple sensor modalities, in order to benefit from using the fusion to improve the context awareness. For instance, combining the knowledge about when user is exercising with location information from the floor-based indoor location system, a smart home environment can control smart appliances to make the atmosphere more convenient, such as dim the light and play relaxing music in that specific area.

Online-learning The technology readiness level (TRLs) [Con11] is a method to estimate the maturity of a technology. In order to reach the highest readiness level requires far more efforts beyond the prototype stage. Most of the machine learning models trained today use a fixed amount of training data and thus do not generalize well on new data. The ability to cope with new, unseen data, without the need to retrain the model again is thus a critical requirement to build a mature system. The model should possess the ability of curriculum learning. This problem should be targeted by approaches, such as continuous learning or online learning probability. A way is to be found to keep the human-in-the-loop to improve the model. It should be ensured, that if the model is uncertain about certain sample, the decision should be forwarded to a human inspector to label. The new knowledge afterwards should be integrated into the current model to enhance its ability in the future.

Explainable Model Traditional classification algorithms with reasonable amount of feature dimensions allow us to make clear decision boundaries. A decision tree model is able to provide a decision by following simple decision rules. Neural network architectures with millions of parameters are more difficult to reason why a certain decision was made. With the hype of using deep learning methods in sectors such as the military, health care and finance, a fail decision could lead to large damage. For human-centered decision especially, it is even more critical that the decision are explainable. Approaches to make explainable models are one of the demanding research direction in this field. DARPA launched the project *Explainable Artificial Intelligence* (XAI) [Gun17] which aims to create a collection of machine learning tools to help producing more explainable models and enables human users to understand the decision the model has taken. Adding reliability measure to the prediction is also another interesting research topic. IBM launched the project *Trusting AI* [BDH*19] which is another toolkit to detect and mitigate algorithmic bias. Increasing the reliability in a decision made by a model or an algorithm also increases fairness and adds transparency to the decision making process. Therefore, the trend towards explainable AI is the next dominant research direction attracting most of the research experts from different disciplines.

A. Publications and Talks

The thesis is partially based on the following publications and talks:

A.1. Full Conference Papers

1. **Biying Fu**, Florian Kirchbuchner, Arjan Kuijper: **Unconstrained Workout Activity Recognition on Unmodified Commercial off-the-shelf Smartphones.** PETRA 2020: 20:1-20:10
2. **Biying Fu**, Florian Kirchbuchner, Arjan Kuijper: **Data Augmentation for Time Series: Traditional vs Generative Models on Capacitive Proximity Time Series.** PETRA 2020: 16:1-16:10
3. **Biying Fu**, Matthias Ruben Mettel, Florian Kirchbuchner, Andreas Braun, Arjan Kuijper: **Surface Acoustic Arrays to Analyze Human Activities in Smart Environments.** AmI 2018: 115-130
4. **Biying Fu**, Florian Kirchbuchner, Julian von Wilmsdorff, Tobias Grosse-Puppendahl, Andreas Braun, Arjan Kuijper: **Indoor localization based on passive electric field sensing.** AmI 2018: 131-146
5. **Biying Fu**, Dinesh Vaithyalingam Gangatharan, Arjan Kuijper, Florian Kirchbuchner, Andreas Braun: **Exercise Monitoring On Consumer Smart Phones Using Ultrasonic Sensing.** iWOAR 2017: 9:1-9:6
6. **Biying Fu**, Jakob Karolus, Tobias Grosse-Puppendahl, Jonathan Hermann, Arjan Kuijper: **Opportunities for activity recognition using ultrasound Doppler sensing on unmodified mobile phones.** iWOAR 2015: 8:1-8:10
7. **Biying Fu**, Tobias Grosse-Puppendahl, Arjan Kuijper: **A gesture recognition method for proximity-sensing surfaces in smart environments.** HCI (21) 2015: 163-173

A.2. Full Journal Papers

1. **Biying Fu**, Florian Kirchbuchner, Arjan Kuijper: **Performing Realistic Workout Activity Recognition on Consumer Smartphones.** Technologies 2020, 8, 65
2. **Biying Fu**, Naser Damer, Florian Kirchbuchner, Arjan Kuijper: **Sensing Technology for Human Activity Recognition: a Comprehensive Survey.** IEEE Access (Volume: 8) 2020: 83719-83820
3. **Biying Fu**, Lennart Jarms, Florian Kirchbuchner, Arjan Kuijper: **ExerTrack - towards smart surfaces to track exercises.** Technologies 2020, 8(1), 17
4. **Biying Fu**, Florian Kirchbuchner, Julian von Wilmsdorff, Tobias Grosse-Puppendahl, Andreas Braun, Arjan Kuijper: **Performing indoor localization with electric potential sensing.** J. Ambient Intell. Humaniz. Comput. 10(2): 731-746 (2019)
5. **Biying Fu**, Florian Kirchbuchner, Arjan Kuijper, Andreas Braun, Dinesh Vaithyalingam Gangatharan: **Fitness Activity Recognition on Smartphones Using Doppler Measurements.** Informatics 5(2): 24(2018)

A.3. Working Papers

1. **Biying Fu**, Naser Damer, Florian Kirchbuchner, Arjan Kuijper: **Generalization of Fitness Exercise Recognition from Doppler Measurements by Domain-adaption and Few-Shot Learning** (accepted in 25^{th} *International Conference on Pattern Recognition* (2020), Workshop on Deep Learning for Human-Centric Activity Understanding.)

A.4. Other Contributions

1. Olaf Henninger, **Biying Fu**, Cong Chen: **On the assessment of face image quality based on handcrafted features.** BIOSIG 2020: 273-280
2. Julian von Wilmsdorff, Florian Kirchbuchner, **Biying Fu**, Andreas Braun, Arjan Kuijper: **An experimental overview on electric field sensing.** J. Ambient Intell. Humaniz. Comput. 10(2): 813-824 (2019)
3. Dirk Siegmund, Sudeep Dev, **Biying Fu**, Doreen Scheller, Andreas Braun: **A Look at Feet: Recognizing Tailgating via Capacitive Sensing.** HCI (22) 2018: 139-151
4. Lisa Scher, Forian Kirchbuchner, Julian von Wilmsdorff, **Biying Fu**, Andreas Braun, Arjan Kuijper: **Step by Step: Early Detection of Diseases Using an Intelligent Floor.** AmI 2018: 131-146
5. Dirk Siegmund, Timotheos Samartzidis, **Biying Fu**, Andreas Braun, Arjan Kuijper: **Fiber Defect Detection of Inhomogeneous Voluminous Textiles.** MCPR 2017: 278-287
6. Julian von Wilmsdorff, Florian Kirchbuchner, **Biying Fu**, Andreas Braun, Arjan Kuijper: **An exploratory study on electric field sensing.** AmI 2017: 247-262
7. Florian Kirchbuchner., **Biying Fu**, Andreas Braun, Julian von Wilmsdorff: **New Approaches for Localization and Activity Sensing in Smart Environments.** Ambient Assisted Living 2017: 73-84
8. Dirk Siegmund, **Biying Fu**, Timotheos Samartzidis, Aidmar Wainakh, Arjan Kuijper, Andreas Braun: **Attack detection in an autonomous entrance system using optical flow.** ICDP 2016: 1-6
9. Tobias Grosse-Puppendahl, Xavier Dellangnol, Christian Hatzfeld, Biying Fu, Mario Kupnik, Matthias R. Hastall, James Scott, Marco Gruteser: **Platypus: Indoor Localization and Identification through Sensing of Electric Potential Changes in Human Bodies.** MobiSys 2016: 17-30

B. Supervising Activities

The following list summarizes the student bachelor, diploma and master thesis supervised by the author. The results of these works were partially used as an input into the thesis.

B.1. Diploma and Master Thesis

1. Lian, Runze - Anomaly Detection and probable path prediction for Single and Multiperson-Application in Smart Homes - M.Sc. TU Darmstadt 2019
2. Jarms, Lennart - CapMat for Sport Exercise Recognition and Tracking - M.Sc. TU Darmstadt 2018
3. Sah, Ashish Prasad - Human Activity Recognition Using Single Wire Electrode Based on Electric Potential Sensing - M.Sc. TU Darmstadt 2018
4. Gangatharan, Dinesh Vaithyalingam - Activity Recognition On Unmodified Consumer Smartphones Via Active Ultrasonic Sensing - M.Sc. TU Darmstadt 2017
5. Sagare, Anagha - Best Practices to Visualize Activity Data in Mobile Apps - M.Sc. TU Darmstadt 2017
6. Karolus, Jakob - Opportunities and Applications of Ultrasound Sensing on Unmodified Consumer-grade Smartphones -M.Sc. TU Darmstadt 2015
7. Dellangnol, Xavier - Indoor Localization Based on Electric Potential Sensing - M.Sc. TU Darmstadt 2015

B.2. Bachelor Thesis

1. Stoll, Christian - Indoor Localization using Particle Filter approach for Single and Multiperson Application in Smart Home - B.Sc. TU Darmstadt 2019
2. Medina, Francisco - CapFloor - a Smart Floor for Sport Exercise Recognition - B.Sc. TU Darmstadt 2018

C. Curriculum Vitae

Personal Data

Name	Biying Fu
Birth date	21.04.1987
Birth place	Shanghai, China
Nationality	German

Education

2011 – 2014	Master of Science in Electrical Engineering and Information Technology at Karlsruher Institute for Technology, Germany
2008 – 2011	Bachelor of Science in Electrical Engineering and Information Technology at Karlsruher Institute for Technology, Germany

Work Experience

2014 –	Researcher, Smart Living and Biometric Technologies, Fraunhofer Institute for Computer Graphics Research, Darmstadt, Germany, Focus: Sensor applications for human activity recognition in smart environments

Bibliography

[AA01] ALI A., AGGARWAL J. K.: Segmentation and recognition of continuous human activity. In *Proceedings IEEE Workshop on Detection and Recognition of Events in Video* (July 2001), pp. 28–35. 27

[AB10] ALTUN K., BARSHAN B.: Human activity recognition using inertial/magnetic sensor units. In *International workshop on human behavior understanding* (2010), Springer, pp. 38–51. 38

[ACS17] ALZANTOT M., CHAKRABORTY S., SRIVASTAVA M. B.: Sensegen: A deep learning architecture for synthetic sensor data generation. *CoRR abs/1701.08886* (2017). 124

[AGG*13] AUMI M. T. I., GUPTA S., GOEL M., LARSON E., PATEL S.: Doplink: Using the doppler effect for multi-device interaction. In *Proceedings of the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing* (2013), UbiComp '13, ACM, pp. 583–586. 53

[AHD*11] ALFORD A., HANSEN C., DOZIER G. V., BRYANT K. S., KELLY J. C., ABEGAZ T., RICANEK K., WOODARD D. L.: Gec-based multi-biometric fusion. In *Proceedings of the IEEE Congress on Evolutionary Computation, CEC 2011, New Orleans, LA, USA, 5-8 June, 2011* (2011), IEEE, pp. 2071–2074. 40

[AJLS97] ADDLESEE M. D., JONES A., LIVESEY F., SAMARIA F.: The orl active floor [sensor system]. *IEEE Personal Communications 4*, 5 (Oct 1997), 35–41. 26, 28, 45

[AK13] ADIB F., KATABI D.: See through walls with wifi! In *Proceedings of the ACM SIGCOMM 2013 Conference on SIGCOMM* (New York, NY, USA, 2013), SIGCOMM '13, Association for Computing Machinery, pp. 75–86. 16, 36, 37, 39, 49

[ARK*06] ALWAN M., RAJENDRAN P., KELL S., MACK D., DALAL S., WOLFE M., FELDER R.: A smart and passive floor-vibration based fall detector for elderly. vol. 1, pp. 1003 – 1007. 16, 17, 19, 49

[ATSK12] ALBERT M. V., TOLEDO S., SHAPIRO M., KOERDING K.: Using mobile phones for activity recognition in parkinson's patients. *Frontiers in neurology 3* (2012), 158. 45

[AWA*15] ALMASSRI A. M., WAN HASAN W. Z., AHMAD S. A., ISHAK A. J., GHAZALI A. M., TALIB D. N., WADA C.: Pressure Sensor: State of the Art, Design, and Application for Robotic Hand. *Journal of Sensors 2015*, 1 (2015), 1–12. 26

[AY09] AHN H.-S., YU W.: Environmental-adaptive rssi-based indoor localization. *IEEE Transactions on Automation Science and Engineering 6*, 4 (2009), 626–633. 2

[AYH15] ABDELNASSER H., YOUSSEF M., HARRAS K. A.: Wigest: A ubiquitous wifi-based gesture recognition system. In *2015 IEEE Conference on Computer Communications (INFOCOM)* (2015), IEEE, pp. 1472–1480. 16, 36, 37, 39, 45, 49

[BBC02] BELLOT D., BOYER A., CHARPILLET F.: A new definition of qualified gain in a data fusion process: application to telemedicine. In *Proceedings of the Fifth International Conference on Information Fusion. FUSION 2002. (IEEE Cat.No.02EX5997)* (July 2002), vol. 2, pp. 865–872 vol.2. 40

[BBS14] BULLING A., BLANKE U., SCHIELE B.: A tutorial on human activity recognition using body-worn inertial sensors. *ACM Computing Surveys (CSUR) 46*, 3 (2014), 1–33. 1

[BDH*19] BELLAMY R. K., DEY K., HIND M., HOFFMAN S. C., HOUDE S., KANNAN K., LOHIA P., MARTINO J., MEHTA S., MOJSILOVIĆ A., ET AL.: Ai fairness 360: An extensible toolkit for detecting and mitigating algorithmic bias. *IBM Journal of Research and Development 63*, 4/5 (2019), 4–1. 202

[BDKK19] BOUTROS F., DAMER N., KIRCHBUCHNER F., KUIJPER A.: Eye-mms: Miniature multi-scale segmentation network of key eye-regions in embedded applications. In *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)* (Oct 2019), pp. 3665–3670. 31

[BFMW15] BRAUN A., FRANK S., MAJEWSKI M., WANG X.: Capseat: Capacitive proximity sensing for automotive activity recognition. In *Proceedings of the 7th International Conference on Automotive User Interfaces and Interactive Vehicular Applications* (New York, NY, USA, 2015), AutomotiveUI '15, Association for Computing Machinery, pp. 225–232. 21, 23, 25, 49, 97

[BHH*13] BRÄNZEL A., HOLZ C., HOFFMANN D., SCHMIDT D., KNAUST M., LÜHNE P., MEUSEL R., RICHTER S., BAUDISCH P.: Gravityspace: Tracking users and their poses in a smart room using a pressure-sensing floor. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2013), CHI '13, Association for Computing Machinery, pp. 725–734. 2, 16, 26, 28, 49

[BHW11] BRAUN A., HEGGEN H., WICHERT R.: Capfloor–a flexible capacitive indoor localization system. In *International Competition on Evaluating AAL Systems through Competitive Benchmarking* (2011), Springer, pp. 26–35. 2, 96, 126, 178

[BI04] BAO L., INTILLE S. S.: Activity recognition from user-annotated acceleration data. In *Pervasive* (2004), Ferscha A., Mattern F., (Eds.), vol. 3001 of *Lecture Notes in Computer Science*, Springer, pp. 1–17. 38

[BKK15] BRAUN A., KREPP S., KUIJPER A.: Acoustic tracking of hand activities on surfaces. In *Proceedings of the 2Nd International Workshop on Sensor-based Activity Recognition and Interaction* (New York, NY, USA, 2015), iWOAR '15, ACM, pp. 9:1–9:5. 177

[BKVM15] BOUTHILLIER X., KONDA K., VINCENT P., MEMISEVIC R.: Dropout as data augmentation. *arXiv preprint arXiv:1506.08700* (2015). 123

[BLO*05] BORRIELLO G., LIU A., OFFER T., PALISTRANT C., SHARP R.: Walrus: Wireless acoustic location with room-level resolution using ultrasound. In *Proceedings of the 3rd International Conference on Mobile Systems, Applications, and Services* (New York, NY, USA, 2005), MobiSys '05, ACM, pp. 191–203. 53

[BS02] BALASUBRAMANIAN M., SCHWARTZ E. L.: The isomap algorithm and topological stability. *Science 295*, 5552 (2002), 7–7. 128

[BY13] BARSHAN B., YUKSEK M.: Recognizing daily and sports activities in two open source machine learning environments using body-worn sensor units. *The Computer Journal 57* (10 2013), 1649–1667. 41, 42

[BZP14] BANNIS A., ZHANG P., PAN S.: Adding directional context to gestures using doppler effect. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication* (New York, NY, USA, 2014), UbiComp '14 Adjunct, ACM, pp. 5–8. 65

[C*15] CHOLLET F., ET AL.: Keras, 2015. 110, 154

[CBHK02] CHAWLA N. V., BOWYER K. W., HALL L. O., KEGELMEYER W. P.: Smote: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research 16* (2002), 321–357. 124, 177

[CBNKL18] CHRIST M., BRAUN N., NEUFFER J., KEMPA-LIEHR A. W.: Time Series FeatuRe Extraction on basis of Scalable Hypothesis tests (tsfresh – A Python package). *Neurocomputing 307* (2018), 72–77. 108, 154

[CCY14] CHERNBUMROONG S., CANG S., YU H.: Genetic algorithm-based classifiers fusion for multisensor activity recognition of elderly people. *IEEE journal of biomedical and health informatics 19*, 1 (2014), 282–289. 40, 42

[CD04] COOK D., DAS S. K.: *Smart environments: technology, protocols, and applications*, vol. 43. John Wiley & Sons, 2004. 2

[CFK13] COOK D., FEUZ K. D., KRISHNAN N. C.: Transfer learning for activity recognition: A survey. *Knowledge and information systems 36*, 3 (2013), 537–556. 7

[CGGS16] CIPPITELLI E., GASPARRINI S., GAMBI E., SPINSANTE S.: A human activity recognition system using skeleton data from rgbd sensors. *Computational intelligence and neuroscience 2016* (2016). 30, 31

[CGL*12] COHN G., GUPTA S., LEE T.-J., MORRIS D., SMITH J. R., REYNOLDS M. S., TAN D. S., PATEL S. N.: An ultra-low-power human body motion sensor using static electric field sensing. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing* (New York, NY, USA, 2012), UbiComp '12, ACM, pp. 99–102. 23

[CHN*12] CHEN L., HOEY J., NUGENT C. D., COOK D. J., YU Z.: Sensor-based activity recognition. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews) 42*, 6 (2012), 790–808. 1

[CHS*18] CAO Z., HIDALGO G., SIMON T., WEI S.-E., SHEIKH Y.: OpenPose: realtime multi-person 2D pose estimation using Part Affinity Fields. In *arXiv preprint arXiv:1812.08008* (2018). 27

[CLG17] CHENG L., LI Y., GUAN Y.: Human activity recognition based on compressed sensing. In *2017 IEEE 7th annual computing and communication workshop and conference (CCWC)* (2017), IEEE, pp. 1–7. 44, 45

[CMPT12] COHN G., MORRIS D., PATEL S., TAN D.: Humantenna: Using the body as an antenna for real-time whole-body interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2012), CHI '12, Association for Computing Machinery, pp. 1901–1910. 16, 23, 25, 49

[CNW11] CHEN L., NUGENT C. D., WANG H.: A knowledge-driven approach to activity recognition in smart homes. *IEEE Transactions on Knowledge and Data Engineering 24*, 6 (2011), 961–974. 1

[Coh95] COHEN L.: *Time-frequency analysis*, vol. 778. Prentice hall, 1995. 56

[Con11] CONROW E. H.: Estimating technology readiness level coefficients. *Journal of Spacecraft and Rockets 48*, 1 (2011), 146–152. 202

[CSZ*16] CHENG J., SUNDHOLM M., ZHOU B., HIRSCH M., LUKOWICZ P.: Smart-surface: Large scale textile pressure sensors arrays for activity recognition. *Pervasive and Mobile Computing 30* (2016), 97–112. 16, 24, 26, 49

[CYL*17] CHENG L., YU Y., LIU X., SU J., GUAN Y.: Recognition of human activities using fast and adaptive sparse representation based on wearable sensors. In *2017 16th IEEE International Con-*

ference on Machine Learning and Applications (ICMLA) (2017), IEEE, pp. 944–949. 44, 45

[Dam18] DAMER N.: *Application-driven Advances in Multi-biometric Fusion*. PhD thesis, Darmstadt University of Technology, Germany, 2018. 40

[Das99] DASGUPTA S.: Learning mixtures of gaussians. In *40th Annual Symposium on Foundations of Computer Science (Cat. No. 99CB37039)* (1999), IEEE, pp. 634–644. 28

[DBM*19] DAMER N., BOUTROS F., MALLAT K., KIRCHBUCHNER F., DUGELAY J., KUIJPER A.: Cascaded generation of high-quality color visible face images from thermal captures. *CoRR abs/1910.09524* (2019). 31

[DD15] DODONOV V., DODONOV A.: Energy–time and frequency–time uncertainty relations: Exact inequalities. *Physica Scripta 90*, 7 (2015), 074049. 79

[DDS16] DAS DAWN D., SHAIKH S. H.: A comprehensive survey of human action recognition with spatio-temporal interest point (stip) detector. *Vis. Comput. 32*, 3 (Mar. 2016), 289–306. 29

[Die] DIETER B.: Google's project soli: The tech behind pixel 4's motion sense radar. *The VERGE*. 34

[Die16] DIENEL G. A.: In memoriam louis sokoloff, m.d. 1921–2015. *Journal of Cerebral Blood Flow & Metabolism 36*, 2 (2016), 278–280. PMID: 26661214. 13

[DLDW12] DERPANIS K. G., LECCE M., DANIILIDIS K., WILDES R. P.: Dynamic scene understanding: The role of orientation features in space and time in scene classification. In *2012 IEEE Conference on Computer Vision and Pattern Recognition* (June 2012), pp. 1306–1313. 29

[DlTHM*09] DE LA TORRE F., HODGINS J., MONTANO J., VALCARCEL S., FORCADA R., MACEY J.: Guide to the carnegie mellon university multimodal activity (cmu-mmac) database. *Robotics Institute, Carnegie Mellon University 5* (2009). 41, 42

[DO14] DAMER N., OPEL A.: Multi-biometric score-level fusion and the integration of the neighbors distance ratio. In *Image Analysis and Recognition - 11th International Conference, ICIAR 2014, Vilamoura, Portugal, October 22-24, 2014, Proceedings, Part II* (2014), Campilho A. J. C., Kamel M. S., (Eds.), vol. 8815 of *Lecture Notes in Computer Science*, Springer, pp. 85–93. 40

[DON14] DAMER N., OPEL A., NOUAK A.: Biometric source weighting in multi-biometric fusion: Towards a generalized and robust solution. In *22nd European Signal Processing Conference, EUSIPCO 2014, Lisbon, Portugal, September 1-5, 2014* (2014), IEEE, pp. 1382–1386. 40

[DP16] DADI H. S., PILLUTLA G. M.: Improved face recognition rate using hog features and svm classifier. *IOSR Journal of Electronics and Communication Engineering 11*, 04 (2016), 34–44. 8

[DPG16] DIBA A., PAZANDEH A. M., GOOL L. V.: Efficient two-stream motion and appearance 3d cnns for video classification. *CoRR abs/1608.08851* (2016). 29

[DRBK17] DAMER N., RHAIBANI C. I., BRAUN A., KUIJPER A.: Trust the biometrie mainstream: Multi-biometric fusion and score coherence. In *25th European Signal Processing Conference, EUSIPCO 2017, Kos, Greece, August 28 - September 2, 2017* (2017), IEEE, pp. 2191–2195. 40

[DSB99] DI STEFANO L., BULGARELLI A.: A simple and efficient connected components labeling algorithm. In *Proceedings 10th International Conference on Image Analysis and Processing* (1999), IEEE, pp. 322–327. 30

[DT01] DOCKSTADE S. L., TEKALP A. M.: Multiple camera tracking of interacting and occluded human motion. *Proceedings of the IEEE 89*, 10 (Oct 2001), 1441–1455. 30, 33

[DWW15] DU Y., WANG W., WANG L.: Hierarchical recurrent neural network for skeleton based action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (2015), pp. 1110–1118. 31

[DZG*18] DING C., ZHANG L., GU C., BAI L., LIAO Z., HONG H., LI Y., ZHU X.: Non-contact human motion recognition based on uwb radar. *IEEE Journal on Emerging and Selected Topics in Circuits and Systems 8*, 2 (2018), 306–315. 35

[ETHC91] ELBLE R. J., THOMAS S. S., HIGGINS C., COLLIVER J.: Stride-dependent changes in gait of older people. *Journal of Neurology 238*, 1 (1991), 1–5. 189

[FAKA*18] FRID-ADAR M., KLANG E., AMITAI M., GOLDBERGER J., GREENSPAN H.: Synthetic data augmentation using gan for improved liver lesion classification. In *2018 IEEE 15th international symposium on biomedical imaging (ISBI 2018)* (2018), IEEE, pp. 289–293. 123

[FCC10] FILONENKO V., CULLEN C., CARSWELL J.: Investigating ultrasonic positioning on mobile phones. In *Indoor Positioning and Indoor Navigation (IPIN), 2010 International Conference on* (Sept 2010), pp. 1–8. 52

[FDKK20a] FU B., DAMER N., KIRCHBUCHNER F., KUIJPER A.: Generalization of fitness exercise recognition from doppler measurements by domain-adaption and few-shot learning. In 25^{th} *International Conference on Pattern Recognition* (2020), Workshop on Deep Learning for Human-Centric Activity Understanding. 138

[FDKK20b] FU B., DAMER N., KIRCHBUCHNER F., KUIJPER A.: Sensing technology for human activity recognition: A comprehensive survey. *IEEE Access 8* (2020), 83791–83820. 13

[FFW*18] FAWAZ H. I., FORESTIER G., WEBER J., IDOUMGHAR L., MULLER P.-A.: Data augmentation using synthetic data for time series classification with deep residual networks. *ArXiv abs/1808.02455* (2018). 123

[FGPK15] FU B., GROSSE-PUPPENDAHL T., KUIJPER A.: A gesture recognition method for proximity-sensing surfaces in smart environments. In *International Conference on Distributed, Ambient, and Pervasive Interactions* (2015), Springer, pp. 163–173. 156

[Fic06] FICKER T.: Electrification of human body by walking. *Journal of Electrostatics 64*, 1 (2006), 10–16. 179

[FJKK20] FU B., JARMS L., KIRCHBUCHNER F., KUIJPER A.: Exertrack—towards smart surfaces to track exercises. *Technologies 8*, 1 (2020), 17. 99

[FKGP*15] FU B., KAROLUS J., GROSSE-PUPPENDAHL T., HERMANN J., KUIJPER A.: Opportunities for activity recognition using ultrasound doppler sensing on unmodified mobile phones. In *Proceedings of the 2nd international Workshop on Sensor-based Activity Recognition and Interaction* (2015), ACM, p. 8. 57, 67

[FKK*18] FU B., KIRCHBUCHNER F., KUIJPER A., BRAUN A., VAITHYALINGAM GANGATHARAN D.: Fitness activity recognition on smartphones using doppler measurements. In *Informatics* (2018), vol. 5, Multidisciplinary Digital Publishing Institute, p. 24. 67

[FKK20a] FU B., KIRCHBUCHNER F., KUIJPER A.: Data augmentation for time series: traditional vs generative models on capacitive proximity time series. In *Proceedings of the 13th ACM International Conference on PErvasive Technologies Related to Assistive Environments* (2020), pp. 1–10. 122

[FKK20b] FU B., KIRCHBUCHNER F., KUIJPER A.: Unconstrained workout activity recognition on unmodified commercial off-the-shelf smartphones. In *Proceedings of the 13th ACM International Conference on PErvasive Technologies Related to Assistive Environments* (2020), pp. 1–10. 77,

142

[FKvW*18] FU B., KIRCHBUCHNER F., VON WILMSDORFF J., GROSSE-PUPPENDAHL T., BRAUN A., KUIJPER A.: Performing indoor localization with electric potential sensing. *Journal of Ambient Intelligence and Humanized Computing* (06 2018). 16, 22, 23, 25, 49, 180

[FM11] FAGGION L., MAHDI A. E.: Noncontact human electrophysiological measurements using a new displacement current sensor. In *SENSORS, 2011 IEEE* (Oct 2011), pp. 296–299. 22, 179

[FMB*16] FENG G., MAI J., BAN Z., GUO X., WANG G.: Floor pressure imaging for fall detection with fiber-optic sensors. *IEEE Pervasive Computing 15*, 2 (Apr 2016), 40–47. 26, 28, 49

[FMK*18] FU B., METTEL M. R., KIRCHBUCHNER F., BRAUN A., KUIJPER A.: Surface acoustic arrays to analyze human activities in smart environments. In *European Conference on Ambient Intelligence* (2018), Springer, pp. 115–130. 166

[FOL16] FILIPPOUPOLITIS A., OLIFF W., LOUKAS G.: *Occupancy Detection for Building Emergency Management Using BLE Beacons*. Springer International Publishing, Cham, 2016, pp. 233–240. 178

[FRL05] FOLSTER F., ROHLING H., LUBBERT U.: An automotive radar network based on 77 ghz fmcw sensors. In *IEEE International Radar Conference, 2005.* (May 2005), pp. 871–876. 32

[FZ07] FASTL H., ZWICKER E.: *Psychoacoustics: Facts and Models*. Springer series in information sciences. Springer, 2007. 55

[GCC*19] GHOSH A., CHAKRABORTY A., CHAKRABORTY D., SAHA M., SAHA S.: Ultrasense: A non-intrusive approach for human activity identification using heterogeneous ultrasonic sensor grid for smart home environment. *Journal of Ambient Intelligence and Humanized Computing* (03 2019), 1–22. 8, 17, 45

[GCP*18] GHOSH A., CHAKRABORTY D., PRASAD D., SAHA M., SAHA S.: Can we recognize multiple human group activities using ultrasonic sensors? pp. 557–560. 18, 19, 20

[GH88] GOLDBERG D. E., HOLLAND J. H.: Genetic algorithms and machine learning. 40

[GHP11] GONG N.-W., HODGES S., PARADISO J. A.: Leveraging conductive inkjet technology to build a scalable and versatile surface for ubiquitous sensing. In *Proceedings of the 13th international conference on Ubiquitous computing* (2011), pp. 45–54. 126

[Gir15] GIRSHICK R.: Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision* (2015), pp. 1440–1448. 31

[GJM13] GRAVES A., JAITLY N., MOHAMED A.-R.: Hybrid speech recognition with deep bidirectional lstm. In *2013 IEEE workshop on automatic speech recognition and understanding* (2013), IEEE, pp. 273–278. 84

[Gli00] GLINSKY A.: *Theremin: ether music and espionage*. University of Illinois Press, 2000. 96

[GMPT12] GUPTA S., MORRIS D., PATEL S., TAN D.: Soundwave: Using the doppler effect to sense gestures. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (2012), CHI '12, ACM, pp. 1911–1914. 53, 56

[GP] GROSSE-PUPPENDAHL T.: OpenCapSense: A Rapid Prototyping Toolkit for Pervasive Interaction Using Capacitive Sensing: 18 - 22 March 2013, San Diego, USA. In *2013 IEEE International Conference on Pervasive Computing and Communications (PerCom), San Diego (18–22 March 2013)*. pp. 152–159. 100

[GPBB*13] GROSSE-PUPPENDAHL T., BERGHOEFER Y., BRAUN A., WIMMER R., KUIJPER A.: Opencapsense: A rapid prototyping toolkit for pervasive interaction using capacitive sensing. In *Pervasive Computing and Communications (PerCom), 2013 IEEE International Conference on* (March 2013), pp. 152–159. 97

[GPBW14] GROSSE-PUPPENDAHL T., BECK S., WILBERS D.: Rainbowfish: Visual feedback on gesture-recognizing surfaces. In *CHI '14 Extended Abstracts on Human Factors in Computing Systems* (New York, NY, USA, 2014), CHI EA '14, ACM, pp. 427–430. 153, 156

[GPDH*16] GROSSE-PUPPENDAHL T., DELLANGNOL X., HATZFELD C., FU B., KUPNIK M., KUIJPER A., HASTALL M., SCOTT J., GRUTESER M.: Platypus - indoor localization and identification through sensing electric potential changes in human bodies. In *Proceedings of the 14th Annual International Conference on Mobile Systems, Applications, and Services* (New York, NY, USA, 2016), MobiSys '16, ACM. 16, 23, 25, 49

[GRM14] GAGLIO S., RE G. L., MORANA M.: Human activity recognition process using 3-d posture data. *IEEE Transactions on Human-Machine Systems 45*, 5 (2014), 586–597. 30

[Grö01] GRÖCHENIG K.: *The Short-Time Fourier Transform*. Birkhäuser Boston, Boston, MA, 2001, pp. 37–58. 8, 169

[GSC*17] GHOSH A., SANYAL A., CHAKRABORTY A., SHARMA P., SAHA M., NANDI S., SAHA S.: On automatizing recognition of multiple human activities using ultrasonic sensor grid. pp. 488–491. 8, 16, 17, 45, 49

[GSD*13] GUPTA J. P., SINGH N., DIXIT P., SEMWAL V. B., DUBEY S. R.: Human activity recognition using gait pattern. *International Journal of Computer Vision and Image Processing (IJCVIP) 3*, 3 (2013), 31–53. 30, 33, 49

[Gun17] GUNNING D.: Explainable artificial intelligence (xai). *Defense Advanced Research Projects Agency (DARPA), nd Web 2* (2017). 202

[HB05] HAN J., BHANU B.: Human activity recognition in thermal infrared imagery. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)-Workshops* (2005), IEEE, pp. 17–17. 30

[HFH*09] HALL M., FRANK E., HOLMES G., PFAHRINGER B., REUTEMANN P., WITTEN I. H.: The weka data mining software: An update. *SIGKDD Explor. Newsl. 11*, 1 (Nov. 2009), 10–18. 170

[HGDG17a] HE K., GKIOXARI G., DOLLÁR P., GIRSHICK R.: Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision* (2017), pp. 2961–2969. 8, 31

[HGDG17b] HE K., GKIOXARI G., DOLLÁR P., GIRSHICK R. B.: Mask R-CNN. *CoRR abs/1703.06870* (2017). 27

[HN10] HOLM S., NILSEN C.-I. C.: Robust ultrasonic indoor positioning using transmitter arrays. In *Indoor Positioning and Indoor Navigation (IPIN), 2010 International Conference on* (2010), IEEE, pp. 1–5. 178

[HO00] HYVÄRINEN A., OJA E.: Independent component analysis: Algorithms and applications. *Neural Netw. 13*, 4-5 (May 2000), 411–430. 178

[Hoy18] HOY M. B.: Alexa, siri, cortana, and more: an introduction to voice assistants. *Medical reference services quarterly 37*, 1 (2018), 81–88. 15

[HS93] HASSIBI B., STORK D. G.: Second order derivatives for network pruning: Optimal brain surgeon. In *Advances in neural information processing systems* (1993), pp. 164–171. 32

[HSA*16] HAROON A., SHAH M. A., ASIM Y., NAEEM W., KAMRAN M., JAVAID Q.: Constraints in the iot: the world in 2020 and beyond. *Constraints 7*, 11 (2016), 252–271. 1

[HVD15] HINTON G., VINYALS O., DEAN J.: Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531* (2015). 10

[HWL*03] HELAL S., WINKLER B., LEE C., KADDOURA Y., RAN L., GIRALDO C., KUCHIBHOTLA S., MANN W.: Enabling location-aware pervasive computing applications for the elderly. In *Pervasive Computing and Communications, 2003. (PerCom 2003). Proceedings of the First IEEE International Conference on* (March 2003), pp. 531–536. 53

[HWP*14] HEVESI P., WILLE S., PIRKL G., WEHN N., LUKOWICZ P.: Monitoring household activities and user location with a cheap, unobtrusive thermal sensor array. In *Proceedings of the 2014 ACM international joint conference on pervasive and ubiquitous computing* (2014), pp. 141–145. 30

[HWZ*19] HUANG C., WU X., ZHANG X., LIN S., CHAWLA N. V.: Deep prototypical networks for imbalanced time series classification under data scarcity. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management* (2019), pp. 2141–2144. 140

[IKH13] IBARA K., KANETSUNA K., HIRAKAWA M.: *Identifying Individuals' Footsteps Walking on a Floor Sensor Device*. Springer International Publishing, Cham, 2013, pp. 56–63. 181, 189

[ILT*06] INTILLE S., LARSON K., TAPIA E., BEAUDIN J., KAUSHIK P., NAWYN J., ROCKINSON R.: Using a live-in laboratory for ubiquitous computing research. vol. 3968, pp. 349–365. 41, 42, 43

[IMTP12] IOSIFIDIS A., MARAMI E., TEFAS A., PITAS I.: Eating and drinking activity recognition based on discriminant analysis of fuzzy distances and activity volumes. In *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (2012), IEEE, pp. 2201–2204. 45

[IYA16] IAN GOODFELLOW, YOSHUA BENGIO, AARON COURVILLE: *Deep Learning*. MIT Press, 2016. 105, 122

[JGC80] JACK M. A., GRANT P. M., COLLINS J. H.: The theory, design, and applications of surface acoustic wave fourier-transform processors. *Proceedings of the IEEE 68*, 4 (April 1980), 450–468. 167

[JHJP08] JOON-HO LIM, HYUNCHUL JANG, JAEWON JANG, PARK S.: Daily activity recognition system for the elderly using pressure sensors. In *2008 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society* (Aug 2008), pp. 5188–5191. 16, 26, 28, 49

[JLP06] JIN G.-Y., LU X.-Y., PARK M.-S.: An indoor localization mechanism using active rfid tag. In *IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing (SUTC'06)* (2006), vol. 1, IEEE, pp. 4–pp. 2

[KAY*18] KHURANA R., AHUJA K., YU Z., MANKOFF J., HARRISON C., GOEL M.: Gymcam: Detecting, recognizing and tracking simultaneous exercises in unconstrained scenes. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies 2* (12 2018), 1–17. 1, 16, 30, 33, 49

[KBK04] KIM K. H., BANG S. W., KIM S. R.: Emotion recognition system using short-term monitoring of physiological signals. *Medical and biological engineering and computing 42*, 3 (2004), 419–427. 1

[KGPH*15] KIRCHBUCHNER F., GROSSE-PUPPENDAHL T., HASTALL M. R., DISTLER M., KUIJPER A.: Ambient intelligence from senior citizens' perspectives: Understanding privacy concerns, technology acceptance, and expectations. In *European Conference on Ambient Intelligence* (2015),

Springer, pp. 48–59. 178

[KJvdS17] KAYALIBAY B., JENSEN G., VAN DER SMAGT P.: Cnn-based segmentation of medical imaging data. *arXiv preprint arXiv:1701.03056* (2017). 27

[KKI*17] KAWASHIMA T., KAWANISHI Y., IDE I., MURASE H., DEGUCHI D., AIZAWA T., KAWADE M.: Action recognition from extremely low-resolution thermal image sequence. In *2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)* (2017), IEEE, pp. 1–6. 31, 33

[KL51] KULLBACK S., LEIBLER R. A.: On information and sufficiency. *Ann. Math. Statist. 22*, 1 (03 1951), 79–86. 130

[KL07] KUNZE K., LUKOWICZ P.: Symbolic object localization through active sampling of acceleration and sound signatures. In *Proceedings of the 9th International Conference on Ubiquitous Computing* (Berlin, Heidelberg, 2007), UbiComp '07, Springer-Verlag, pp. 163–180. 53

[KM15] KIM Y., MOON T.: Human detection and activity classification based on micro-doppler signatures using deep convolutional neural networks. *IEEE geoscience and remote sensing letters 13*, 1 (2015), 8–12. 35, 37, 39, 45

[KR08] KALGAONKAR K., RAJ B.: Recognizing talking faces from acoustic doppler reflections. In *Automatic Face Gesture Recognition, 2008. FG '08. 8th IEEE International Conference on* (Sept 2008), pp. 1–6. 53

[KR09] KALGAONKAR K., RAJ B.: One-handed gesture recognition using ultrasonic doppler sonar. In *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on* (April 2009), pp. 1889–1892. 52

[KRM18] KHAN M. A. A. H., ROY N., MISRA A.: Scaling human activity recognition via deep learning-based domain adaptation. In *2018 IEEE International Conference on Pervasive Computing and Communications (PerCom)* (2018), IEEE, pp. 1–9. 139

[KSH07] KE Y., SUKTHANKAR R., HEBERT M.: Spatio-temporal shape and flow correlation for action recognition. In *2007 IEEE conference on computer vision and pattern recognition* (2007), IEEE, pp. 1–8. 45

[KSH12] KRIZHEVSKY A., SUTSKEVER I., HINTON G. E.: ImageNet Classification with Deep Convolutional Neural Networks. In *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1* (USA, 2012), NIPS'12, Curran Associates Inc, pp. 1097–1105. 29, 83, 105, 123

[KTL*13] KE S.-R., THUC H. L. U., LEE Y.-J., HWANG J.-N., YOO J.-H., CHOI K.-H.: A review on video-based human activity recognition. *Computers 2*, 2 (2013), 88–131. 7

[KTS*14] KARPATHY A., TODERICI G., SHETTY S., LEUNG T., SUKTHANKAR R., FEI-FEI L.: Large-scale video classification with convolutional neural networks. In *2014 IEEE Conference on Computer Vision and Pattern Recognition* (June 2014), pp. 1725–1732. 29, 41, 43

[KW52] KRUSKAL W., WALLIS W.: Use of ranks in one-criterion variance analysis. *Journal of the American Statistical Association* (1952), 583–621. 187

[KW13] KINGMA D. P., WELLING M.: Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114* (2013). 128

[KWM11a] KWAPISZ J. R., WEISS G. M., MOORE S. A.: Activity Recognition using Cell Phone Accelerometers. *ACM SIGKDD Explorations Newsletter 12*, 2 (2011), 74. 2

[KWM11b] KWAPISZ J. R., WEISS G. M., MOORE S. A.: Activity recognition using cell phone accelerometers. *ACM SigKDD Explorations Newsletter 12*, 2 (2011), 74–82. 140

[KY55] KUHN H. W., YAW B.: The hungarian method for the assignment problem. *Naval Res. Logist. Quart* (1955), 83–97. 186

[KZS15] KOCH G., ZEMEL R., SALAKHUTDINOV R.: Siamese neural networks for one-shot image recognition. In *ICML deep learning workshop* (2015), vol. 2, Lille. 140

[L*00] LOGAN B., ET AL.: Mel frequency cepstral coefficients for music modeling. In *Ismir* (2000), vol. 270, pp. 1–11. 35

[Lan02] LANGHEINRICH M.: A Privacy Awareness System for Ubiquitous Computing Environments. In *UbiComp 2002: Ubiquitous Computing* (Berlin, Heidelberg, 2002), Borriello G., Holmquist L. E., (Eds.), Springer Berlin Heidelberg, pp. 237–245. 32

[LBG11] LI N., BECERIK-GERBER B.: Performance-based evaluation of rfid-based indoor location sensing solutions for the built environment. *Adv. Eng. Inform. 25*, 3 (aug 2011), 535–546. 178

[LBH15] LECUN Y., BENGIO Y., HINTON G. E.: Deep learning. *Nature 521*, 7553 (2015), 436–444. 122

[LCSL18] LIYANAGE M., CHANG C., SRIRAMA S., LOKE S.: Indoor people density sensing using wi-fi and channel state information. *Advances in modelling and analysis A 61*, 1 (2018), 37–47. 34

[LGK*16] LIEN J., GILLIAN N., KARAGOZLER M. E., AMIHOOD P., SCHWESIG C., OLSON E., RAJA H., POUPYREV I.: Soli: Ubiquitous gesture sensing with millimeter wave radar. *ACM Transactions on Graphics (TOG) 35*, 4 (2016), 142. 16, 34, 37, 39, 49

[LHJ19] LI X., HE Y., JING X.: A survey of deep learning-based human activity recognition in radar. *Remote Sensing 11*, 9 (2019), 1068. 7

[LHL*13] LEE H. J., HWANG S. H., LEE S. M., LIM Y. G., PARK K. S.: Estimation of body postures on bed using unconstrained ecg measurements. *IEEE Journal of Biomedical and Health Informatics 17*, 6 (Nov 2013), 985–993. 21, 25, 49

[LJ18] L. JARMS B. FU A. K.: *CapMat for Sport Exercise Recognition and Tracking*. Tech. rep., Technische Universit at Darmstadt, Fraunhoferstrasse 5, 64283 Darmstadt, October 2018. 125, 126, 127, 155, 157

[LL06] LU W.-L., LITTLE J. J.: Simultaneous tracking and action recognition using the pca-hog descriptor. In *Proceedings of the The 3rd Canadian Conference on Computer and Robot Vision* (USA, 2006), CRV '06, IEEE Computer Society, p. 6. 27, 49

[LL10] LIU R., LIU M.: Recognizing human activities based on multi-sensors fusion. In *2010 4th International Conference on Bioinformatics and Biomedical Engineering* (2010), IEEE, pp. 1–4. 38

[LL13] LARA O. D., LABRADOR M. A.: A Survey on Human Activity Recognition using Wearable Sensors. *IEEE Communications Surveys & Tutorials 15*, 3 (2013), 1192–1209. 2, 7

[LLO04] LU X., LIU Q., OE S.: Recognizing non-rigid human actions using joints tracking in space-time. In *Proceedings of the International Conference on Information Technology: Coding and Computing (ITCC'04) Volume 2 - Volume 2* (USA, 2004), ITCC '04, IEEE Computer Society, p. 620. 27, 49

[LLP*14] LI H., LI Y., PORIKLI F., ET AL.: Deeptrack: Learning discriminative feature representations by convolutional neural networks for visual tracking. In *BMVC* (2014), vol. 1, p. 3. 120

[LMT16] LE GUENNEC A., MALINOWSKI S., TAVENARD R.: Data Augmentation for Time Series Classification using Convolutional Neural Networks. In *ECML/PKDD Workshop on Advanced Analytics and Learning on Temporal Data* (Riva Del Garda, Italy, 2016). 105

[LN05] LAI C.-P., NARAYANAN R. M.: Through-wall imaging and characterization of human activity using ultrawideband (uwb) random noise radar. In *Sensors, and Command, Control, Communications, and Intelligence (C3I) Technologies for Homeland Security and Homeland Defense IV* (2005), vol. 5778, International Society for Optics and Photonics, pp. 186–195. 35, 37

[LPRS12] LIU L., POPESCU M., RANTZ M., SKUBIC M.: Fall detection using doppler radar and classifier fusion. In *Proceedings of 2012 IEEE-EMBS International Conference on Biomedical and Health Informatics* (2012), IEEE, pp. 180–183. 35, 39, 49

[LSWL12] LI X., SHEN M., WANG W., LIU H.: Real-time sound source localization for a mobile robot based on the guided spectral-temporal position method. *International Journal of Advanced Robotic Systems 9*, 3 (2012), 78. 168

[LWH03] LUO Y., WU T.-D., HWANG J.-N.: Object-based analysis and interpretation of human motion in sports video sequences by dynamic bayesian networks. *Computer Vision and Image Understanding 92* (11 2003), 196–216. 27, 49

[LWNS07] LIM C. H., WAN Y., NG B. P., SEE C. M. S.: A real-time indoor wifi localization system utilizing smart antennas. *IEEE Transactions on Consumer Electronics 53*, 2 (May 2007), 618–622. 178

[LWX*19] LI W., WANG L., XU J., HUO J., GAO Y., LUO J.: Revisiting local descriptor based image-to-class measure for few-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2019), pp. 7260–7268. 148

[LZL10] LI W., ZHANG Z., LIU Z.: Action recognition based on a bag of 3d points. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops* (2010), IEEE, pp. 9–14. 30

[MAP*15] MARTIN ABADI, ASHISH AGARWAL, PAUL BARHAM, EUGENE BREVDO, ZHIFENG CHEN, CRAIG CITRO, GREG S. CORRADO, ANDY DAVIS, JEFFREY DEAN, MATTHIEU DEVIN, SANJAY GHEMAWAT, IAN GOODFELLOW, ANDREW HARP, GEOFFREY IRVING, MICHAEL ISARD, YANGQING JIA, RAFAL JOZEFOWICZ, LUKASZ KAISER, MANJUNATH KUDLUR, JOSH LEVENBERG, DANDELION MANÉ, RAJAT MONGA, SHERRY MOORE, DEREK MURRAY, CHRIS OLAH, MIKE SCHUSTER, JONATHON SHLENS, BENOIT STEINER, ILYA SUTSKEVER, KUNAL TALWAR, PAUL TUCKER, VINCENT VANHOUCKE, VIJAY VASUDEVAN, FERNANDA VIÉGAS, ORIOL VINYALS, PETE WARDEN, MARTIN WATTENBERG, MARTIN WICKE, YUAN YU, XIAOQIANG ZHENG: TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems, 2015. 110

[MASB19] METTEL M. R., ALEKSEEW M., STOCKLÖW C., BRAUN A.: Designing and evaluating safety services using depth cameras. *Journal of Ambient Intelligence and Humanized Computing 10*, 2 (2019), 747–759. 29, 31

[MCT*13] MUJIBIYA A., CAO X., TAN D. S., MORRIS D., PATEL S. N., REKIMOTO J.: The sound of touch: On-body touch and gesture sensing based on transdermal ultrasound propagation. In *Proceedings of the 2013 ACM International Conference on Interactive Tabletops and Surfaces* (2013), ITS '13, ACM, pp. 189–198. 53

[MKRMM18] MISHRA A., KRISHNA REDDY S., MITTAL A., MURTHY H. A.: A generative model for zero shot learning using conditional variational autoencoders. In *Proceedings of the IEEE Conference*

on Computer Vision and Pattern Recognition Workshops (2018), pp. 2188–2196. 124

[MM17] MISHRA A., MARR D.: Apprentice: Using knowledge distillation techniques to improve low-precision network accuracy. *arXiv preprint arXiv:1711.05852* (2017). 32

[MNR92] MEHROTRA R., NAMUDURI K. R., RANGANATHAN N.: Gabor filter-based edge detection. *Pattern recognition 25*, 12 (1992), 1479–1494. 8

[MP03] MOTA S., PICARD R. W.: Automated posture analysis for detecting learner's interest level. In *2003 Conference on Computer Vision and Pattern Recognition Workshop* (2003), vol. 5, IEEE, pp. 49–49. 26, 28

[MPC16] MEHR H. D., POLAT H., CETIN A.: Resident activity recognition in smart homes by using artificial neural networks. In *2016 4th international istanbul smart grid congress and fair (ICSG)* (2016), IEEE, pp. 1–5. 45

[MPK09] MESSING R., PAL C., KAUTZ H.: Activity recognition using the velocity histories of tracked keypoints. In *2009 IEEE 12th international conference on computer vision* (2009), IEEE, pp. 104–111. 41, 43

[MPZN16] MIRSHEKARI M., PAN S., ZHANG P., NOH H. Y.: Characterizing wave propagation to improve indoor step-level person localization using floor vibration. In *Sensors and Smart Structures Technologies for Civil, Mechanical, and Aerospace Systems 2016* (2016), Lynch J. P., (Ed.), vol. 9803, International Society for Optics and Photonics, SPIE, pp. 30 – 40. 17, 20, 49

[MRC*07] MÜLLER M., RÖDER T., CLAUSEN M., EBERHARDT B., KRÜGER B., WEBER A.: Documentation mocap database hdm05. 41

[MSSD06] MAURER U., SMAILAGIC A., SIEWIOREK D. P., DEISHER M.: Activity recognition and monitoring using multiple sensors on different body positions. In *Proceedings of the International Workshop on Wearable and Implantable Body Sensor Networks* (USA, 2006), BSN '06, IEEE Computer Society, p. 113–116. 38

[MSU17] MATTHIES D., STRECKER B., URBAN B.: Earfieldsensing: A novel in-ear electric field sensing to enrich wearable gesture input through facial expressions. 23

[MVPEL18] MARTÍNEZ-VILLASEÑOR L., PONCE H., ESPINOSA-LOERA R. A.: Multimodal database for human activity recognition and fall detection. In *Multidisciplinary Digital Publishing Institute Proceedings* (2018), vol. 2, p. 1237. 8, 45

[N*11] NG A., ET AL.: Sparse autoencoder. *CS294A Lecture notes 72*, 2011 (2011), 1–19. 29

[NDHC10] NAZERFARD E., DAS B., HOLDER L. B., COOK D. J.: Conditional random fields for activity recognition in smart environments. In *Proceedings of the 1st ACM International Health Informatics Symposium* (2010), ACM, pp. 282–286. 45

[Ng17] NG A.: Machine learning yearning. *URL: http://www. mlyearning. org/(96)* (2017). 139

[NGW15] NANDAKUMAR R., GOLLAKOTA S., WATSON N.: Contactless sleep apnea detection on smartphones. In *Proceedings of the 13th Annual International Conference on Mobile Systems, Applications, and Services* (New York, NY, USA, 2015), MobiSys '15, Association for Computing Machinery, pp. 45–57. 16, 18, 19, 20, 49, 53, 65

[NH10] NAIR V., HINTON G. E.: Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)* (2010), pp. 807–814. 76

[NHA*00] NAKAMURA S., HIYANE K., ASANO F., NISHIURA T., YAMADA T.: Acoustical sound database in real environments for sound scene understanding and hands-free speech recognition.

In *LREC* (2000). 16

[NITG16] NANDAKUMAR R., IYER V., TAN D., GOLLAKOTA S.: Fingerio : Using sonar for fine-grained finger tracking. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (2016), CHI '16, ACM, p. to appear. 16, 18, 19, 20, 49

[NN08] NATARAJAN P., NEVATIA R.: View and scale invariant action recognition using multiview shape-flow models. In *2008 IEEE Conference on Computer Vision and Pattern Recognition* (2008), IEEE, pp. 1–8. 43

[NP00] NEE R. V., PRASAD R.: *OFDM for wireless multimedia communications*. Artech House, Inc., 2000. 32

[NSN*10] NISHIYAMA M., SASAKI H., NOSE S., TAKAMI K., WATANABE K.: Smart pressure sensing mats with embedded hetero-core fiber optic nerve sensors. *Materials and Manufacturing Processes 25*, 4 (2010), 264–267. 26, 28

[NSY15] NGUYEN T. V., SONG Z., YAN S.: Stap: Spatial-temporal attention-aware pooling for action recognition. *IEEE Transactions on Circuits and Systems for Video Technology 25*, 1 (Jan 2015), 77–86. 29, 33

[OA00] ORR R., ABOWD G.: The smart floor: A mechanism for natural user identification and tracking. 27, 45

[Ors12] ORSLEY T. J.: Capacitive touchscreen or touchpad for finger and active stylus, Oct. 2 2012. US Patent 8,278,571. 23

[PBJ*14] PAN S., BONDE A., JING J., ZHANG L., ZHANG P., NOH H. O.: Boes: building occupancy estimation system using sparse ambient vibration monitoring. In *Sensors and Smart Structures Technologies for Civil, Mechanical, and Aerospace Systems 2014* (2014), vol. 9061, International Society for Optics and Photonics, p. 90611O. 166

[PBRW*08] PRANCE R., BEARDSMORE-RUST S., WATSON P., HARLAND C., PRANCE H.: Remote detection of human electrophysiological signals using electric potential sensors. *Applied Physics Letters 93* (07 2008), 033906–033906. 16, 22, 23, 25, 49

[PCD15] PINHEIRO P. O., COLLOBERT R., DOLLÁR P.: Learning to segment object candidates. In *Advances in Neural Information Processing Systems* (2015), pp. 1990–1998. 31

[PFKP05] PATTERSON D., FOX D., KAUTZ H., PHILIPOSE M.: Fine-grained activity recognition by aggregating abstract object usage. vol. 2005, pp. 44 – 51. 40, 41

[PGF*16] POUPYREV I., GONG N.-W., FUKUHARA S., KARAGOZLER M. E., SCHWESIG C., ROBINSON K. E.: Project jacquard: Interactive digital textiles at scale. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2016), CHI '16, Association for Computing Machinery, p. 4216–4227. 97

[PGGP13] PU Q., GUPTA S., GOLLAKOTA S., PATEL S.: Whole-home gesture recognition using wireless signals. In *Proceedings of the 19th Annual International Conference on Mobile Computing Networking* (New York, NY, USA, 2013), MobiCom '13, Association for Computing Machinery, pp. 27–38. 36, 37, 39

[PGH*16] PU Y., GAN Z., HENAO R., YUAN X., LI C., STEVENS A., CARIN L.: Variational autoencoder for deep learning of images, labels and captions. In *Advances in Neural Information Processing Systems 29*, Lee D. D., Sugiyama M., Luxburg U. V., Guyon I., Garnett R., (Eds.). Curran Associates, Inc., 2016, pp. 2352–2360. 128

[PGM*19] PASZKE A., GROSS S., MASSA F., LERER A., BRADBURY J., CHANAN G., KILLEEN T., LIN Z., GIMELSHEIN N., ANTIGA L., DESMAISON A., KOPF A., YANG E., DEVITO Z., RAISON M., TEJANI A., CHILAMKURTHY S., STEINER B., FANG L., BAI J., CHINTALA S.: Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, Wallach H., Larochelle H., Beygelzimer A., dAlché-Buc F., Fox E., Garnett R., (Eds.). Curran Associates, Inc., 2019, pp. 8024–8035. 82

[PHO11] PLÖTZ T., HAMMERLA N. Y., OLIVIER P. L.: Feature learning for activity recognition in ubiquitous computing. In *Twenty-Second International Joint Conference on Artificial Intelligence* (2011). 45

[PK13] PIYATHILAKA L., KODAGODA S.: Gaussian mixture based hmm for human daily activity recognition using 3d skeleton features. In *2013 IEEE 8th conference on industrial electronics and applications (ICIEA)* (2013), IEEE, pp. 567–572. 8, 45

[Pop10] POPPE R.: A survey on vision-based human action recognition. *Image and vision computing 28*, 6 (2010), 976–990. 43

[PPA18] PATEL A., PRABHUDESAI C., AKSANLI B.: Non-intrusive activity detection and prediction in smart residential spaces. In *2018 9th IEEE Annual Ubiquitous Computing, Electronics Mobile Communication Conference (UEMCON)* (Nov 2018), pp. 355–361. 18

[PPPR16] POURYAZDAN A., PRANCE R. J., PRANCE H., ROGGEN D.: Wearable electric potential sensing: A new modality sensing hair touch and restless leg movement. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct* (New York, NY, USA, 2016), UbiComp '16, Association for Computing Machinery, pp. 846–850. 23

[PSZ12] PENG C., SHEN G., ZHANG Y.: Beepbeep: A high-accuracy acoustic-based system for ranging and localization using cots devices. *ACM Trans. Embed. Comput. Syst. 11*, 1 (Apr. 2012), 4:1–4:29. 53

[PVG*11] PEDREGOSA F., VAROQUAUX G., GRAMFORT A., MICHEL V., THIRION B., GRISEL O., BLONDEL M., PRETTENHOFER P., WEISS R., DUBOURG V., VANDERPLAS J., PASSOS A., COURNAPEAU D., BRUCHER M., PERROT M., DUCHESNAY E.: Scikit-learn: Machine learning in python. *J. Mach. Learn. Res. 12* (nov 2011), 2825–2830. 69

[PWQ*15] PAN S., WANG N., QIAN Y., VELIBEYOGLU I., NOH H. Y., ZHANG P.: Indoor person identification through footstep induced structural vibration. In *Proceedings of the 16th International Workshop on Mobile Computing Systems and Applications* (New York, NY, USA, 2015), HotMobile '15, ACM, pp. 81–86. 17, 19, 20, 49, 166

[QHX*14] QIFAN Y., HAO T., XUEBING Z., YIN L., SANFENG Z.: Dolphin: Ultrasonic-based gesture recognition on smartphone platform. In *Computational Science and Engineering (CSE), 2014 IEEE 17th International Conference on* (Dec 2014), pp. 1461–1468. 16, 18, 19, 20, 49, 53

[QVF18] QASSIM H., VERMA A., FEINZIMER D.: Compressed residual-vgg16 cnn model for big data places image recognition. In *2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC)* (2018), IEEE, pp. 169–175. 82

[QZK08] QIAN G., ZHANG J., KIDANÉ A.: People identification using gait via floor pressure sensing and analysis. In *Smart Sensing and Context* (Berlin, Heidelberg, 2008), Roggen D., Lombriser C., Tröster G., Kortuem G., Havinga P., (Eds.), Springer Berlin Heidelberg, pp. 83–98. 26, 28, 45, 49

[RAAS12] ROHRBACH M., AMIN S., ANDRILUKA M., SCHIELE B.: A database for fine grained activity detection of cooking activities. In *2012 IEEE Conference on Computer Vision and Pattern*

Recognition (2012), IEEE, pp. 1194–1201. 41, 42

[RBD*09] RUCKELSHAUSEN A., BIBER P., DORNA M., GREMMES H., KLOSE R., LINZ A., RAHE F., RESCH R., THIEL M., TRAUTZ D., ET AL.: Bonirob: an autonomous field robot platform for individual plant phenotyping. *Precision agriculture 9*, 841 (2009), 1. 1

[RC16] RONAO C. A., CHO S.-B.: Human activity recognition with smartphone sensors using deep learning neural networks. *Expert Syst. Appl. 59*, C (oct 2016), 235–244. 73

[RCR*10] ROGGEN D., CALATRONI A., ROSSI M., HOLLECZEK T., FORSTER K., TROSTER G., LUKOWICZ P., BANNACH D., PIRKL G., FERSCHA A., DOPPLER J., HOLZMANN C., KURZ M., HOLL G., CHAVARRIAGA R., SAGHA H., BAYATI H., CREATURA M., MILLAN J. D. R.: Collecting complex activity datasets in highly rich networked sensor environments. pp. 233 – 240. 41, 43

[RDGF16] REDMON J., DIVVALA S., GIRSHICK R., FARHADI A.: You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (2016), pp. 779–788. 8, 31

[RDML05] RAVI N., DANDEKAR N., MYSORE P., LITTMAN M. L.: Activity recognition from accelerometer data. In *Aaai* (2005), vol. 5, pp. 1541–1546. 140

[RGPK14] RUS S., GROSSE-PUPPENDAHL T., KUIJPER A.: Recognition of bed postures using mutual capacitance sensing. In *Ambient Intelligence* (Cham, 2014), Aarts E., de Ruyter B., Markopoulos P., van Loenen E., Wichert R., Schouten B., Terken J., Van Kranenburg R., Den Ouden E., O'Hare G., (Eds.), Springer International Publishing, pp. 51–66. 21, 23, 25, 49

[RHGS15] REN S., HE K., GIRSHICK R., SUN J.: Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems* (2015), pp. 91–99. 31

[RHR10] ROHLING H., HEUEL S., RITTER H.: Pedestrian detection procedure integrated into an 24 ghz automotive radar. In *2010 IEEE Radar Conference* (May 2010), pp. 1229–1232. 32

[RI05] REINHART K.-F., ILLING M.: *Automotive Sensor Market*. John Wiley Sons, Ltd, 2005, ch. 2, pp. 5–19. 13

[RJ86] RABINER L., JUANG B.: An introduction to hidden markov models. *ieee assp magazine 3*, 1 (1986), 4–16. 8

[RKH10] RAO K. R., KIM D. N., HWANG J.-J.: *Fast Fourier Transform - Algorithms and Applications*, 1st ed. Springer Publishing Company, Incorporated, 2010. 8, 169

[RKHD12] RAJ B., KALGAONKAR K., HARRISON C., DIETZ P.: Ultrasonic doppler sensing in hci. *Pervasive Computing, IEEE 11*, 2 (Feb 2012), 24–29. 52

[RLBL*18] RAHMAN A., LUBECKE V. M., BORIC-LUBECKE O., PRINS J. H., SAKAMOTO T.: Doppler radar techniques for accurate respiration characterization and subject identification. *IEEE Journal on Emerging and Selected Topics in Circuits and Systems 8*, 2 (2018), 350–359. 16, 34, 39, 45, 49

[RM01] RANDELL C., MULLER H. L.: Low cost indoor positioning system. In *Proceedings of the 3rd International Conference on Ubiquitous Computing* (2001), UbiComp '01, Springer-Verlag, pp. 42–48. 53

[RMC15] RADFORD A., METZ L., CHINTALA S.: Unsupervised representation learning with deep convolutional generative adversarial networks, 2015. cite arxiv:1511.06434Comment: Under review as a conference paper at ICLR 2016. 29

[RMD07] REYNOLDS M., MAZALEK A., DAVENPORT G.: An acoustic position sensing system for large scale interactive displays. In *Sensors, 2007 IEEE* (Oct 2007), pp. 1193–1196. 53

[RMW14] REZENDE D. J., MOHAMED S., WIERSTRA D.: Stochastic backpropagation and approximate inference in deep generative models. In *Proceedings of the 31st International Conference on Machine Learning* (Bejing, China, 22–24 Jun 2014), Xing E. P., Jebara T., (Eds.), vol. 32 of *Proceedings of Machine Learning Research*, PMLR, pp. 1278–1286. 128

[RR15] RANSING R. S., RAJPUT M.: Smart home for elderly care, based on wireless sensor network. In *2015 International Conference on Nascent Technologies in the Engineering Field (ICNTE)* (2015), IEEE, pp. 1–5. 38

[RRR18] RAMASAMY RAMAMURTHY S., ROY N.: Recent trends in machine learning for human activity recognition—a survey. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery 8*, 4 (2018), e1254. 7

[RS12] REISS A., STRICKER D.: Introducing a new benchmarked dataset for activity monitoring. In *2012 16th International Symposium on Wearable Computers* (June 2012), pp. 108–109. 41, 42

[RSA*13] ROSSI M., SEITER J., AMFT O., BUCHMEIER S., TRÖSTER G.: Roomsense: An indoor positioning system for smartphones using active sound probing. In *Proceedings of the 4th Augmented Human International Conference* (2013), AH '13, ACM, pp. 89–95. 53

[RW04] REKIMOTO J., WANG H.: Sensing gamepad: electrostatic potential sensing for enhancing entertainment oriented interactions. pp. 1457–1460. 23, 25

[RW12] RICHARDSON M., WALLACE S.: *Getting started with raspberry PI*. " O'Reilly Media, Inc.", 2012. 98

[RXS10] ROFOUEI M., XU W., SARRAFZADEH M.: Computing with uncertainty in a smart textile surface for object recognition. In *2010 IEEE Conference on Multisensor Fusion and Integration* (Sep. 2010), pp. 174–179. 24, 28, 49

[RZH05] RUBIO J. P. B., ZHOU C., HERNÁNDEZ F. S.: *Vision-Based Walking Parameter Estimation for Biped Locomotion Imitation*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2005, pp. 677–684. 189

[SA19] SHELKE S., AKSANLI B.: Static and dynamic activity detection with ambient sensors in smart spaces. *Sensors 19*, 4 (2019), 804. 8, 16, 30, 33, 45, 49

[SAMM19] SLIM S., ATIA A., M.A. M., MOSTAFA M.-S.: Survey on human activity recognition based on acceleration data. *International Journal of Advanced Computer Science and Applications 10* (01 2019). 7

[SAZ19] SEIFERT A.-K., AMIN M. G., ZOUBIR A. M.: Toward unobtrusive in-home gait analysis based on radar micro-doppler signatures. *IEEE Transactions on Biomedical Engineering 66*, 9 (2019), 2629–2640. 34, 37, 39

[SBQK05] SRINIVASAN P., BIRCHFIELD D., QIAN G., KIDANÉ A.: A pressure sensing floor for interactive media applications. In *Proceedings of the 2005 ACM SIGCHI International Conference on Advances in Computer Entertainment Technology* (New York, NY, USA, 2005), ACE '05, Association for Computing Machinery, pp. 278–281. 16, 26, 28, 49

[SBTM08] SAXENA S., BRÉMOND F., THONNAT M., MA R.: Crowd behavior recognition for video surveillance. In *Advanced Concepts for Intelligent Vision Systems* (Berlin, Heidelberg, 2008), Blanc-Talon J., Bourennane S., Philips W., Popescu D., Scheunders P., (Eds.), Springer Berlin Heidelberg, pp. 970–981. 1, 27, 33, 49

[Sch00] SCHMIDT A.: Implicit human computer interaction through context. *Personal technologies 4*, 2-3 (2000), 191–199. 198

[SCLW17] SHAFIEE M. J., CHYWL B., LI F., WONG A.: Fast yolo: A fast you only look once system for real-time embedded object detection in video. *arXiv preprint arXiv:1709.05943* (2017). 31

[SCZ*14] SUNDHOLM M., CHENG J., ZHOU B., SETHI A., LUKOWICZ P.: Smart-Mat: Recognizing and Counting Gym Exercises with Low-cost Resistive Pressure Sensing Matrix. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing - UbiComp '14 Adjunct* (New York, New York, USA, 2014), Brush A. J., Friday A., Kientz J., Scott J., Song J., (Eds.), ACM Press, pp. 373–382. 1, 16, 24, 28, 45, 49, 110, 111, 115, 116

[SF17] SCHNEIDER A., FEUSSNER H.: Chapter 11 - tracking and navigation systems. In *Biomedical Engineering in Gastrointestinal Surgery*, Schneider A., Feussner H., (Eds.). Academic Press, 2017, pp. 443 – 472. 35

[SFC*11] SHOTTON J., FITZGIBBON A., COOK M., SHARP T., FINOCCHIO M., MOORE R., KIPMAN A., BLAKE A.: Real-time human pose recognition in parts from single depth images. In *Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition* (USA, 2011), CVPR '11, IEEE Computer Society, pp. 1297–1304. 49

[SH17] SAGAYAM K. M., HEMANTH D. J.: Hand posture and gesture recognition techniques for virtual reality applications: a survey. *Virtual Reality 21*, 2 (Jun 2017), 91–107. 27

[Sho15] A Survey of Online Activity Recognition Using Mobile Phones. *Sensors (Basel, Switzerland) 15*, 1 (2015), 2059–2085. 101

[SHS01] SAVVIDES A., HAN C.-C., STRIVASTAVA M. B.: Dynamic fine-grained localization in ad-hoc networks of sensors. In *Proceedings of the 7th Annual International Conference on Mobile Computing and Networking* (2001), MobiCom '01, ACM, pp. 166–179. 53

[SKBM*97] STOLZE H., KUHTZ-BUSCHBECK J., MONDWURF C., BOCZEK-FUNCKE A., JÖHNK K., DEUSCHL G., ILLERT M.: Gait analysis during treadmill and overground locomotion in children and adults. *Electroencephalography and Clinical Neurophysiology/Electromyography and Motor Control 105*, 6 (1997), 490–497. 195

[SKJS15] SCHULZ A., KAROLUS J., JANSSEN F., SCHWEIZER I.: Accurate pollutant modeling and mapping: Applying machine learning to participatory sensing and urban topology data. In *International Conference on Networked Systems (NetSys2015)* (2015). 65

[SKvW*18] SCHERF L., KIRCHBUCHNER F., VON WILMSDORFF J., FU B., BRAUN A., KUIJPER A.: Step by step: Early detection of diseases using an intelligent floor. In *European Conference on Ambient Intelligence* (2018), Springer, pp. 131–146. 24, 193

[SLC04] SCHULDT C., LAPTEV I., CAPUTO B.: Recognizing human actions: a local svm approach. In *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.* (2004), vol. 3, IEEE, pp. 32–36. 41, 43

[SLCS19] SUN Q., LIU Y., CHUA T.-S., SCHIELE B.: Meta-transfer learning for few-shot learning. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (June 2019). 140

[SLY15] SOHN K., LEE H., YAN X.: Learning structured output representation using deep conditional generative models. In *Advances in Neural Information Processing Systems 28*, Cortes C., Lawrence N. D., Lee D. D., Sugiyama M., Garnett R., (Eds.). Curran Associates, Inc., 2015, pp. 3483–3491. 131

[SM06] SAFARIC S., MALARIC K.: Zigbee wireless standard. In *Proceedings ELMAR 2006* (2006), IEEE, pp. 259–262. 38

[Smi96] SMITH J. R.: Field mice: Extracting hand geometry from electric field measurements. *IBM Systems Journal 35* (1996), 587–608. 21, 98

[SN11] SAAD S. S., NAKAD Z. S.: A standalone rfid indoor positioning system using passive tags. *IEEE Transactions on Industrial Electronics 58*, 5 (2011), 1961 – 1970. 178

[SNC*08] SKORDOULIS D., NI Q., CHEN H.-H., STEPHENS A. P., LIU C., JAMALIPOUR A.: Ieee 802.11 n mac frame aggregation mechanisms for next-generation high-throughput wlans. *IEEE Wireless Communications 15*, 1 (2008), 40–47. 32

[SNR*15] SINGH G., NELSON A., ROBUCCI R., PATEL C., BANERJEE N.: Inviz: Low-power personalized gesture recognition using wearable textile capacitive sensor arrays. In *2015 IEEE international conference on pervasive computing and communications (PerCom)* (2015), IEEE, pp. 198–206. 126

[SPBZ13] SUN Z., PUROHIT A., BOSE R., ZHANG P.: Spartacus: Spatially-aware interaction for mobile devices through energy-efficient audio sensing. In *Proceeding of the 11th Annual International Conference on Mobile Systems, Applications, and Services* (2013), MobiSys '13, ACM, pp. 263–276. 53

[SPH12] SATO M., POUPYREV I., HARRISON C.: Touché: Enhancing touch interaction on humans, screens, liquids, and everyday objects. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2012), CHI '12, Association for Computing Machinery, pp. 483–492. 21, 25, 96

[SPSS12] SUNG J., PONCE C., SELMAN B., SAXENA A.: Unstructured human activity detection from rgbd images. In *2012 IEEE international conference on robotics and automation* (2012), IEEE, pp. 842–849. 30

[SSSA12] STORK J. A., SPINELLO L., SILVA J., ARRAS K. O.: Audio-based human activity recognition using non-markovian ensemble voting. In *2012 IEEE RO-MAN: The 21st IEEE International Symposium on Robot and Human Interactive Communication* (2012), IEEE, pp. 509–514. 45

[SSSY17] SIKARWAR S., SATYENDRA, SINGH S., YADAV B. C.: Review on pressure sensors for structural health monitoring. *Photonic Sensors 7*, 4 (2017), 294–304. 24

[SSZ17] SNELL J., SWERSKY K., ZEMEL R.: Prototypical networks for few-shot learning. In *Advances in neural information processing systems* (2017), pp. 4077–4087. 140, 147

[STS*13] SOUSA M., TECHMER A., STEINHAGE A., LAUTERBACH C., LUKOWICZ P.: Human tracking and identification using a sensitive floor and wearable accelerometers. In *2013 IEEE International Conference on Pervasive Computing and Communications (PerCom)* (March 2013), pp. 166–171. 16, 21, 23, 25, 49, 96, 178

[SVB*13] SEIDENARI L., VARANO V., BERRETTI S., BIMBO A., PALA P.: Recognizing actions from depth cameras as weakly aligned multi-part bag-of-poses. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops* (2013), pp. 479–485. 30

[SVLS08] STIKIC M., VAN LAERHOVEN K., SCHIELE B.: Exploring semi-supervised and active learning for activity recognition. In *2008 12th IEEE International Symposium on Wearable Computers* (2008), IEEE, pp. 81–88. 45

[SWvHG11] SCHROEDER J., WABNIK S., VAN HENGEL P. W. J., GOETZE S.: *Detection and Classification of Acoustic Events for In-Home Care*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011,

pp. 181–195. 16, 18, 20, 49

[SYY*13] SHAN X.-J., YIN J.-Y., YU D.-L., LI C.-F., ZHAO J.-J., ZHANG G.-F.: Analysis of artificial corner reflector's radar cross section: a physical optics perspective. *Arabian Journal of Geosciences 6*, 8 (2013), 2755–2765. 67

[SZS12] SOOMRO K., ZAMIR A. R., SHAH M.: UCF101: A dataset of 101 human actions classes from videos in the wild. *CoRR abs/1212.0402* (2012). 29, 41, 43

[TBB09] TENORTH M., BANDOUCH J., BEETZ M.: The tum kitchen data set of everyday manipulation activities for motion tracking and action recognition. In *2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops* (2009), IEEE, pp. 1089–1096. 38, 41, 42

[TBF*15] TRAN D., BOURDEV L., FERGUS R., TORRESANI L., PALURI M.: Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV)* (USA, 2015), ICCV '15, IEEE Computer Society, p. 4489–4497. 29, 45

[TCSU08] TURAGA P., CHELLAPPA R., SUBRAHMANIAN V. S., UDREA O.: Machine recognition of human activities: A survey. *IEEE Transactions on Circuits and Systems for Video technology 18*, 11 (2008), 1473–1488. 43

[TDDM09] TARZIA S. P., DICK R. P., DINDA P. A., MEMIK G.: Sonar-based measurement of user presence and attention. In *Proceedings of the 11th International Conference on Ubiquitous Computing* (2009), Ubicomp '09, ACM, pp. 89–92. 53

[Teo13] TEODORESCU H.-N.: Textile-, conductive paint-based wearable devices for physical activity monitoring. In *2013 E-Health and Bioengineering Conference (EHB)* (2013), IEEE, pp. 1–4. 126

[TH12] TIELEMAN T., HINTON G.: Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural networks for machine learning 4*, 2 (2012), 26–31. 76

[TKY*16] TAHAT A., KADDOUM G., YOUSEFI S., VALAEE S., GAGNON F.: A look at the recent wireless positioning techniques with a focus on algorithms for moving receivers. *IEEE Access 4* (2016), 6652–6680. 16, 36, 37, 39, 49

[TN06] TEMKO A., NADEU C.: Classification of acoustic events using svm-based clustering schemes. *Pattern Recogn. 39*, 4 (Apr. 2006), 682–694. 16, 18, 20, 45, 49

[TSS16] TOLSTIKHIN I. O., SRIPERUMBUDUR B. K., SCHÖLKOPF B.: Minimax estimation of maximum mean discrepancy with radial kernels. In *Advances in Neural Information Processing Systems 29*, Lee D. D., Sugiyama M., Luxburg U. V., Guyon I., Garnett R., (Eds.). Curran Associates, Inc., 2016, pp. 1930–1938. 132

[TY17] TIKHONOV A., YAMSHCHIKOV I. P.: Music generation with variational recurrent autoencoder supported by history. *CoRR abs/1705.05458* (2017). 128

[UNH08] UHRIKOVA Z., NUGENT C. D., HLAVAC V.: The use of computer vision techniques to augment home based sensorised environments. In *2008 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society* (Aug 2008), pp. 2550–2553. 7, 16, 30, 33, 49

[UPP*17] UM T. T., PFISTER F. M. J., PICHLER D., ENDO S., LANG M., HIRCHE S., FIETZEK U., KULIĆ D.: Data Augmentation of Wearable Sensor Data for Parkinson's Disease Monitoring using Convolutional Neural Networks. 216–220. 105, 126

[U.S08] U.S. DEPARTMENT OF HEALTH AND HUMAN SERVICES: Physical Activity Guidelines for Americans. 2

[VKMV11] VALTONEN M., KAILA L., MÄENTAUSTA J., VANHALA J.: Unobtrusive human height and posture recognition with a capacitive sensor. *Journal of Ambient Intelligence and Smart Environments 3*, 4 (2011), 305–332. 125

[vKNEK08] VAN KASTEREN T., NOULAS A., ENGLEBIENNE G., KRÖSE B.: Accurate activity recognition in a home setting. In *Proceedings of the 10th International Conference on Ubiquitous Computing* (New York, NY, USA, 2008), UbiComp '08, ACM, pp. 1–9. 41, 42, 43

[VMV09] VALTONEN M., MAENTAUSTA J., VANHALA J.: Tiletrack: Capacitive human tracking using floor tiles. In *2009 IEEE International Conference on Pervasive Computing and Communications* (March 2009), pp. 1–10. 16, 21, 23, 25, 49, 97, 178

[VYK13] VEAUX C., YAMAGISHI J., KING S.: The voice bank corpus: Design, collection and data analysis of a large regional accent speech database. pp. 1–4. 16

[WCH*19] WANG J., CHEN Y., HAO S., PENG X., HU L.: Deep learning for sensor-based activity recognition: A survey. *Pattern Recognition Letters 119* (2019), 3–11. 7

[WCS*16] WANG A., CHEN G., SHANG C., ZHANG M., LIU L.: Human activity recognition in a smart home environment with stacked denoising autoencoders. In *International conference on web-age information management* (2016), Springer, pp. 29–40. 45

[Wei01] WEIK M. H.: *Nyquist theorem*. Springer US, Boston, MA, 2001, pp. 1127–1127. 167

[WGH07a] WILLIAMS A., GANESAN D., HANSON A.: Aging in place: fall detection and localization in a distributed smart camera network. pp. 892–901. 7, 30, 33

[WGH07b] WILLIAMS A., GANESAN D., HANSON A.: Aging in place: fall detection and localization in a distributed smart camera network. In *Proceedings of the 15th international conference on Multimedia* (2007), ACM, pp. 892–901. 178

[WGLK16] WANG L., GUPTA S., LOH K. J., KOO H. S.: Distributed pressure sensing using carbon nanotube fabrics. *IEEE Sensors Journal 16*, 12 (June 2016), 4663–4664. 26

[WGSM16] WONG S. C., GATT A., STAMATESCU V., MCDONNELL M. D.: Understanding data augmentation for classification: when to warp? In *2016 international conference on digital image computing: techniques and applications (DICTA)* (2016), IEEE, pp. 1–6. 124

[Whi87] WHITE R. M.: A sensor classification scheme. *IEEE Transactions on ultrasonics, ferroelectrics, and frequency control 34*, 2 (1987), 124–126. 13, 14

[WJQ*17] WANG F., JIANG M., QIAN C., YANG S., LI C., ZHANG H., WANG X., TANG X.: Residual attention network for image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2017), pp. 3156–3164. 31

[WKBS07] WIMMER R., KRANZ M., BORING S., SCHMIDT A.: A capacitive sensing toolkit for pervasive activity detection and recognition. In *Fifth Annual IEEE International Conference on Pervasive Computing and Communications (PerCom'07)* (March 2007), pp. 171–180. 97

[WLG11] WARD J. A., LUKOWICZ P., GELLERSEN H. W.: Performance metrics for activity recognition. *ACM Transactions on Intelligent Systems and Technology (TIST) 2*, 1 (2011), 1–23. 44

[WLS16] WANG W., LIU A. X., SHAHZAD M.: Gait recognition using wifi signals. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing* (2016), ACM, pp. 363–373. 36, 39, 45, 49

[WLWY12] WANG J., LIU Z., WU Y., YUAN J.: Mining actionlet ensemble for action recognition with depth cameras. In *2012 IEEE Conference on Computer Vision and Pattern Recognition* (2012), IEEE, pp. 1290–1297. 41, 43

[WLZ*18] WANG T.-C., LIU M.-Y., ZHU J.-Y., TAO A., KAUTZ J., CATANZARO B.: High-resolution image synthesis and semantic manipulation with conditional gans. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (2018), pp. 8798–8807. 124

[WQT15] WANG L., QIAO Y., TANG X.: Action recognition with trajectory-pooled deep-convolutional descriptors. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (June 2015). 29, 33

[WSL*16] WANG S., SONG J., LIEN J., POUPYREV I., HILLIGES O.: Interacting with soli: Exploring fine-grained dynamic gesture recognition in the radio-frequency spectrum. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology* (New York, NY, USA, 2016), UIST '16, Association for Computing Machinery, pp. 851–860. 16, 35, 39, 45, 49

[WTT13] WATANABE H., TERADA T., TSUKAMOTO M.: Ultrasound-based movement sensing, gesture-, and context-recognition. In *Proceedings of the 2013 International Symposium on Wearable Computers* (2013), ISWC '13, ACM, pp. 57–64. 53

[WZ15] WANG S., ZHOU G.: A review on radio based activity recognition. *Digital Communications and Networks 1*, 1 (2015), 20–29. 7

[WZCH18] WANG J., ZHENG V. W., CHEN Y., HUANG M.: Deep transfer learning for cross-domain activity recognition. In *proceedings of the 3rd International Conference on Crowd Science and Engineering* (2018), pp. 1–8. 139

[XCA12] XIA L., CHEN C.-C., AGGARWAL J. K.: View invariant human action recognition using histograms of 3d joints. In *2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops* (2012), IEEE, pp. 20–27. 30

[XHA*13] XU W., HUANG M.-C., AMINI N., HE L., SARRAFZADEH M.: ecushion: A textile pressure sensor array design and calibration for sitting posture analysis. *IEEE Sensors Journal 13*, 10 (2013), 3926–3934. 24, 45, 49

[YNS*15] YANG J., NGUYEN M. N., SAN P. P., LI X. L., KRISHNASWAMY S.: Deep convolutional neural networks on multichannel time series for human activity recognition. In *Twenty-Fourth International Joint Conference on Artificial Intelligence* (2015). 45

[Yos12] YOSHUA BENGIO: Practical Recommendations for Gradient-Based Training of Deep Architectures. *CoRR abs/1206.5533* (2012). 115, 154

[YSC*11] YANG J., SIDHOM S., CHANDRASEKARAN G., VU T., LIU H., CECAN N., CHEN Y., GRUTESER M., MARTIN R. P.: Detecting driver phone use leveraging car speakers. In *Proceedings of the 17th Annual International Conference on Mobile Computing and Networking* (2011), MobiCom '11, ACM, pp. 97–108. 53

[ZBH*17] ZHANG C., BENGIO S., HARDT M., RECHT B., VINYALS O.: Understanding deep learning requires rethinking generalization. 122

[ZGL] ZAFARI F., GKELIAS A., LEUNG K.: A survey of indoor localization systems and technologies. arxiv 2017. *arXiv preprint arXiv:1709.01015*. 37

[ZHX*07] ZHU G., HUANG Q., XU C., RUI Y., JIANG S., GAO W., YAO H.: Trajectory based event tactics analysis in broadcast sports video. In *Proceedings of the 15th ACM International Conference on Multimedia* (New York, NY, USA, 2007), MM '07, Association for Computing Machinery, pp. 58–67. 29, 33

[ZS09] ZHU C., SHENG W.: Human daily activity recognition in robot-assisted living using multi-sensor fusion. In *2009 IEEE International Conference on Robotics and Automation* (2009), IEEE,

pp. 2154–2159. 38

[ZSE17] ZHAO S., SONG J., ERMON S.: Infovae: Information maximizing variational autoencoders. *CoRR abs/1706.02262* (2017). 132

[ZWX*19] ZENG Y., WU D., XIONG J., YI E., GAO R., ZHANG D.: Farsense: Pushing the range limit of wifi-based respiration sensing with csi ratio of two antennas. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies 3*, 3 (2019), 121. 16, 36, 39, 49

[ZWY16] ZHICHENG CUI, WENLIN CHEN, YIXIN CHEN: Multi-Scale Convolutional Neural Networks for Time Series Classification. *CoRR abs/1603.06995* (2016). 107

[ZYH*18] ZHANG Y., YANG C., HUDSON S. E., HARRISON C., SAMPLE A.: Wall++ room-scale interactive and context-aware sensing. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (2018), pp. 1–15. 96