Application of electric field sensing systems in smart environments

dem Fachbereich Informatik
der Technischen Universität Darmstadt
vorzulegende

DISSERTATION

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

M.Sc. Andreas Braun

geboren in Aschaffenburg, Deutschland

Referenten der Arbeit:	Prof. Dr. techn. Dieter W. Fellner Technische Universität Darmstadt Prof. XXX affiliation of Prof. XXX
Tag der Einreichung:	31/01/2014
Tag der mündlichen Prüfung:	

Darmstädter Dissertation
D 17

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 31/01/2014 Andreas Braun

Abstract

Summarize the thesis in 1/2–1 page.

Zusammenfassung

Describe in German in 6–10 pages your thesis. This is compulsory for EN written thesis. Zusammenfassung auf Deutsch.

Contents

List of Figures

List of Tables

1. Introduction

Smart environments are comprised of numerous sensing and computing devices that are supporting a number of users in this environment on performing their tasks. Capacitive sensors are a technology that uses electric fields to sense the presence and certain properties of the human body. In this work I present an overview of this technology, how it can be applied in different relevant application scenarios and based on various prototypes evaluate the particular benefits and limitations of this sensing technology.

1.1. Motivation

In the last decade the way we interact with computing machines has changed in a profound fashion. Today more than one billion people operate a smartphone, enabling ubiquitous access to communication tools, processing power and information. The vision of ubiquitous computing as proposed by Mark Weiser in the early 90s is inching closer to reality [Wei91]. The required technologies of

> "cheap, low-power computers that include equally convenient displays, a network that ties them all together, and software systems implementing ubiquitous applications"

are now existing in the form of smartphones and tablets that are connected to the internet, using high-speed connections such as LTE and web-based services such as Google Now, that combine numerous data sources to provide personalized services.

While the vision and underlying ideas remain similar other names have been used in research, including Pervasive Computing and Ambient Intelligence. The concept has been expanded to not only consider devices that can be directly manipulated, but include determining the situation and reacting based on it. This context-aware computing proposes

> "systems that examine and react to an individual's changing context. Such systems can promote and mediate people's interactions with devices, computers, and other people" [SAW94]

Different forms of context can be distinguished, ranging from location and the actual system state, to different activities or even the current mood of the user. In order to acquire this context, the input-and-output based systems originally proposed by Weiser, are augmented by an ensemble of devices that are very small (dust), coordinate in massive numbers (clay) or are flexible, unobtrusive extensions to everyday objects (fabric) [Pos11]. This devices can be invisibly integrated into our everyday environment and provide sensing capabilities that can be used by sufficiently smart systems. Examples of these devices are microelectromechanical systems (MEMS) or bendable technology, such as OLED screens. The number of computation and sensing devices that we carry with us is growing continuously, yet we want the technology to further disappear, allowing us to focus on the application instead of the underlying technology.

The famous science fiction author Arthur C. Clarke proposed three laws of prediction, the third of which is

> "Any sufficiently advanced technology is indistinguishable from magic." [Cla62]

Capacitive proximity sensing allows us to measure the influence of the human body (or conductive objects in general) on an electric field. While we would not call this technology magic, a peculiarity of electricity is that humans have no specific sensing organs, thus we generally remain unaware of their presence, unless the field strength is very high. Consequently, when interacting with capacitive sensors there is no awareness of what they are sensing unless it is specifically exposed to the user. The technology is well-understood and varieties have become ubiquitous in some areas, such as touch screens. However, there are numerous other applications for this technology, ranging from industrial fluid level and material detection, to presence detection for cars. A particularly interesting domain for this sensing technology are smart environments that provide services based on unobtrusively acquired information about persons currently acting in this environment. There are numerous sensing technologies that provide similar detection capabilities. Looking at the recognition of simple activities, such as standing, walking and lying, cameras and accelerometers can lead to the same result. Accordingly, in order to discuss the use a sensing technology within a specific domain, it is necessary to provide a benchmark that takes into account abstracted sensor properties and different application domains. In this work we will provide a generic benchmark model for different sensor technologies in smart environments and based on this discuss the use of capacitive proximity sensor technologies in this area. We will establish the most suited application domains and provide prototypes to evaluate different aspects.

1.2. Research Challenges

In the past there have been numerous influential works that gave an overview of technologies and applications in smart environments. Cook et al. identified common technologies, frameworks and applications in this domain and give an overview of ongoing research. Poslad specified a more detailed taxonomy of device classes, provides concepts for interaction between humans and environments and gives an overview of intelligent systems. A different category of previous work details the different sensing technologies that are supporting various different applications and give an overview of limitations and benefits. However, so far there has been no work that provides a benchmark that maps different sensor characteristics to applications in smart environments. As it was stated by Cook and Das [CD07]:

> "Finally, a useful goal for the smart environment research community is to define evaluation mechanisms. While performance measures can be defined for each technology within the architecture hierarchy [...], performance measures for entire smart environments still need to be established. This can form the basis of comparative assessments and identify areas that need further investigation."

An intermediate step between evaluating entire environments and low-level technologies is an application-specific benchmarking of systems. Benchmarking as a method allows us to quantify the performance of a specfic process or item and allows a comparison to competing processes or items. It is common to benchmark different technologies according to their features. My proposal is to extend technology-driven benchmarks by adding an application-specific feature weighting. This approach allows to map the same set of features to different applications that have similar requirements that are catered to by divergent technologies.

This method is tested with capacitive proximity sensors in the domain of smart environments. The past few years have seen several emerging trends in computing, ranging from an increased connectivity of devices, driven by the Internet of Things, ubiquitous usage of mobile computing and sensing devices in the form of smartphones and tablets, and novel, natural interaction paradigms, that aim to provide human-machine interaction similar to interpersonal communication means [Val08]. This trend is apparent within smart environments that have an increased demand for unobtrusive sensing options and provide novel interaction methods to their inhabitants, enabling intuitive control of the environment [Pos11]. Driven by improved embedded technology, materials and an increase in computing power, it is possible to provide integrated systems based on capacitive proximity sensing that contribute novel aspects to several of these trends. Based on the proposed benchmarking method I will identify different applications that are particularly suited for capacitive proximity sensors.

In the last years we have created various prototypes in the identified application domains and applied state-of-the art sensor technology and novel algorithms. In this process we are able to provide numerous improvements to previously presented systems that often rely on uniform sensor arrays [Smi96] or require a large number of sensors [Rek02]. Particular topics include the improvement of sparse-sensor systems that aim at maximizing the information acquired by a limited number of sensors, the use of model-based human body sensing methods and heterogeneous capacitive systems that either combine parallel divergent data processing, non-uniform sensor distributions or functional cooperation with other types of sensors.

Based on this it is possible to provide a thorough review of capacitive proximity sensing technology in the domain of smart environments. The benefits and limitations of the technology can be discussed in detail and a set of application guidelines for capacitive proximity sensors in smart environments can be established.

Indoor localization is a base technology within smart environments, enabling a multitude of applications including augmented reality, navigation in large indoor areas, or sports [TCD*00,IHQ04,LBO11]. Some particular challenges in assisted living applications include low budgets, easy installation, privacy preservation and interoperability with other systems in the environment [CK12]. Camera-based systems are very popular in this domain, yet particularly in private settings struggle with user acceptance. A potential solution are smart camera systems that process and abstract the images before they are sent to the network, however they require efficient and robust tracking algorithms for implementation on embedded systems.

1.3. Contributions

In the following I will list briefly and concisely what are the specific contributions provided by this work on a methodological and practical level. They are distinguished into three different groups, the benchmarking model, the prototypes and an indoor localization system:

- Benchmarking model for sensors in smart environments
  - Identification of application domains in smart environments
  - Application-centric benchmarking model for mapping a single set of sensor features to different smart environment applications
  - Identification of applications suitable for capacitive proximity sensors based on the developed benchmarking model
- Smart environment prototypes using capacitive proximity sensors
  - MagicBox prototype enabling expressive single-hand gestural interaction with sparse sensor distribution and machine learning gesture classification
  - CapFloor prototype using a novel layout for floor-based capacitive indoor localization systems, enabling unobtrusive application, easy maintenance and additional services such as fall detection
  - SmartBed prototype using a model-based approach for fitting one or two persons, concurrently detecting sleep phases and breathing rate for occupants
  - Capacitive Chair prototype that uses allows to detect presence, identify users, track different postures, measure breathing rate and enables novel applications for smart offices
  - Active Armrest prototype uses a heterogeneous sensor layout to enable different forms of interaction in automotive environments
  - CapTap prototype combining capacitive sensors and microphones in a table-based interaction device, enabling multi-hand interaction in three dimensions using a multi-level interaction pattern

- Discussion of Limitations and Benefits of capacitive proximity sensors in smart environments
- Presentation of AmbiTrack - a camera-based indoor localization system for smart environments

1.4. Structure of this work

After having identified the research challenges and introduced the topic the related works are specified in Section 2 - *Related Work* - in four categories. The first section gives a background on electric field sensing, including relevant historical work and the physical properties. Additional different sensing categories are outlined, before different electrode considerations and data processing methods are introduced. The second category of related works discusses different applications of capacitive proximity sensors that were created in the last decades, ranging from MIT research in the early 90s, to novel touch classifiers based on different sensing methods. The third category introduces different competing technologies that will be used in the later benchmarking. Finally we give an overview of existing work collecting and grouping applications in smart environments. This will allow us to identify candidate scenarios for capacitive proximity sensors.

Section 3 - *Benchmarking model for sensors in smart environments* - is concerned with the first main contribution of this work, the introduction of an application-specific benchmarking model for sensors in smart environments. In the first part of this section the sensor features relevant for application in smart environments are discussed. Six relevant features and three omitted features are introduced including the rationale of their inclusion in the model. The next part describes the benchmarking model. The application-based feature weighting is introduced, leading to the derivation of the model itself, including the required calculation of an overall score. After that we are using the model to score different examples and link those to relevant projects within the related works. Finally the model is discussed and will be used to identify suitable applications for capacitive proximity sensors in smart environments.

Section 4 - *Application prototypes* - describes the six different prototypes that have been created for the application domains, MagicBox, CapFloor, Capacitive Chair, Active Armrest, SmartBed and CapTap. After giving a general introduction and linking the prototypes to different application scenarios, each system is described in detail, including design rationale with regard to capacitive sensors, the data processing methods used and the evaluations that have been performed to verify the different prototypes.

The knowledge gathered in designing, building and testing the prototypes and using the benchmarking model leads to Section 5 - *Evaluating capacitive proximity sensors in smart environments*, wherein the results are discussed and evaluated. This section has five parts. At first capacitive proximity sensor are classified within the domain of smart environments, discussing applications and findings of the prototypes. In the next part the technology is compared to the other sensor classes introduced in the related works. Afterwards limitations and benefits of the technology are collected and linked to the sensor features and applications. The section concludes with a set of guidelines that may help interested parties in evaluating their application for usage with capacitive proximity sensors and give practical help when applying this technology.

Section 6 - *AmbiTrack indoor localization system for smart environments* - is presenting a camera-based indoor localization system. While this section is not directly concerned with capacitive proximity sensors, indoor localization is an important technology for numerous smart environment applications. AmbiTrack has been created for participating in the EvAAL competition that evaluated different indoor localization systems for Ambient Assisted Living, ranking the systems based on numerous technical features, user experience and openness of implementation. The section will briefly introduce the requirements for indoor localization systems and present the prototype system and results of the EvAAL competition.

The document concludes in Section 7 - *Conclusions and Future Work* - that briefly recapitulates the work and introduces potential future research stemming from this work.

There are x different appendices. Appendix A lists publications and talks. Appendix B lists Master and Bachelor Thesis that were supervised or co-supervised. Appendix C contains a short CV.

2. Related Work

In this section I will describe the most relevant works that inspired this work or are linked to a specific topic. The aim of this section is to provide a basis for both, the benchmarking model that is developed in section 3, and the capacitive proximity sensing prototypes described in section 4. The related works are distinguished into four distinct parts. At first I will give a general introduction to electric field sensing, including a discussion on different properties, physical background, the influence of materials and geometry and different data processing methods. Afterwards I will present relevant applications using capacitive proximity sensing, ranging from historical works to very recent systems. In the next section various sensing technologies are introduced that are used in smart environment systems. Finally I will identify and group different applications in smart environments, providing a basis for the benchmarking model.

2.1. Electric field sensing

Different electric charges apply either a repelling or attracting force to each other. For any point in space these forces have a distinct direction and magnitude. The resulting collection of force vectors is called the electric field. Conductive objects that are present in this area modify the properties of the field. Electric field sensing enables measuring field properties at a certain point in space. Using continuous monitoring it is possible to gather information about conductive objects passing through the field by associating measured disturbances to properties of the object. It is possible to gather a multitude of different information about a project. In this section I will give an overview of the physical background, different measurement modes and how to process data acquired by digital sensors.

2.1.1. Physical properties

A complete overview about the electrostatic principles of capacitive proximity sensing can be found in the book by Baxter [Bax96], chapters 2 and 6. We will give a very brief introduction to this topic in the following section. The basic setup of a typically used sensor is shown in Figure 2.1. The proximity capacitance C_x can be determined

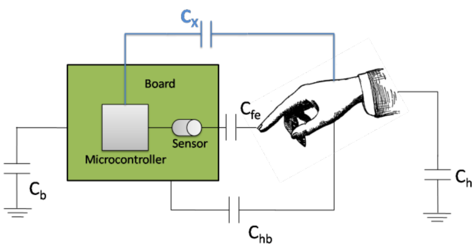

Figure 2.1.: Black box setup of a capacitive proximity sensor

using a combination of serial and parallel circuits of capacitors, resulting in the following equation:

$$C_x = \left(\left(C_{hb} + \frac{C_h C_b}{C_h + C_b} \right)^{-1} \frac{1}{C_{fe}} \right)^{-1} \tag{2.1}$$

Additionally there are parasitic capacitance components, i.e. disturbing capacitance values within the system. Sources are:

- Sensing electrode capacitance
- Capacitance between sensing electrode and ground plane
- Intercapacitance between neighboring traces on the board

The present parasitic capacitances C_{par} amount to values approximately between $10pF$ and $300pF$ and are therefore considerably larger than the value of the proximity capacitance Cx, being between $0.1pF$ and $10pF$. The total capacitance sensed is the sum of parasitic and proximity components.

$$C_S = C_X + C_{par} \tag{2.2}$$

It is obvious that this parasitic capacitance is considerably higher than the capacitance induced by an approaching object. However, this parasitic capacitance is typically static and can therefore be calibrated in a way not affecting the measurement. Now we will shortly discuss how we can estimate the capacitance of common ob-

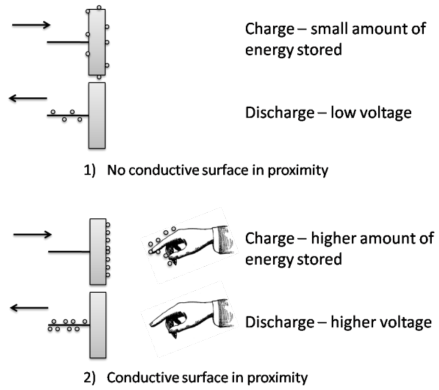

Figure 2.2.: Capacitive sensing procedure

jects that approach the sensor. Any object exhibits capacitance in respect to infinity. Surveying simple geometric shapes this capacitance is analytically determinable, e.g.:

$$C = 8\varepsilon_0 rDisk \tag{2.3}$$

$$C = 4\pi\varepsilon_0 rSphere \tag{2.4}$$

ε_0 is the vacuum permittivity and r the respective radius. This free space capacitance is increasing as soon as another object is approaching, caused by the capacitance of this second object, resulting in mutual capacitance. Looking at generic formulas, determining capacitance between parallel plates this behavior can be described analytically.

$$C = \frac{Q}{V} \qquad C = \varepsilon_0 \varepsilon_r \frac{A}{d} \tag{2.5}$$

The capacitance is directly proportional to the plate area A and inversely proportional to the distance d between the plates, with ε_r being the relative static permittivity of the dielectric between the plates. Sensor electronics are grounded with the body acting as ground itself. The sensor plate is continuously charged using a constant voltage V. A higher capacitance allows the system to hold a larger charge. If the system is connected to the ground, the sensor capacitor is discharged through a resistor. The resulting voltage is depending on the available charge, shown in the equation above. Furthermore the required time to discharge the capacitor is increased. This process is symbolized in Figure 2.2.

2.1.2. Proximity sensing versus touch sensing

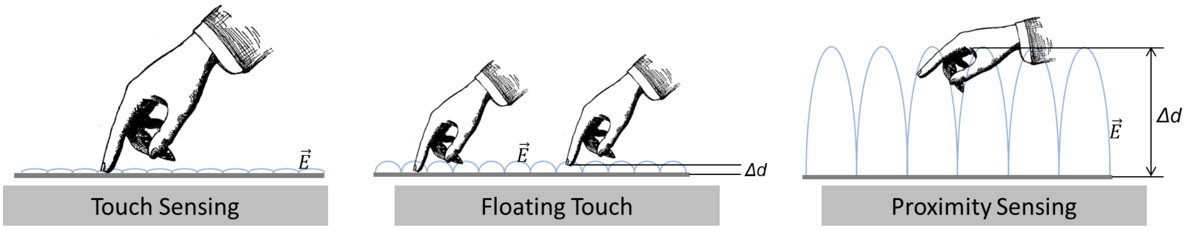

Figure 2.3.: Different projected capacitive sensing methods based on distance

The most ubiquitous usage of capacitive sensing technology can be found in touch screens. As the trend went from pen-controlled mobile systems to finger controlled devices with the first iPhone in 2007, projected capacitance touch is the most prevalent technology for touch screens. It uses various layers of transparent electrodes or nanowires to detect the mutual capacitance as objects enter the detection area [BO10b]. The commercially available devices have gained additional abilities over the last few years, leading to the development of "floating touch" systems that are able to track fingers in gloves or fingers that are hovering above the surface [Cyp12, Nok12]. Applications are the usage of mobile devices in cold outdoor temperatures or additional navigation fea-tures based on the hovering fingers. In consequence we can distinguish the three different projected capacitive sensing methods as shown in Figure 2.3:

- Touch sensing - densely distributed sensors are tuned to project a weak electric field in order to detect one or more objects touching the interactive surface. The sensors have to be close to the surface.
- Floating touch - densely distributed high-sensitivity sensors are able to detect both touches and very near objects ($< 2cm$) to enable usage using protective gear or additional navigation feature. The sensors have to be close to the surface.
- Proximity sensing - sparsely distributed sensors create a stronger electric field that propagates into space in order to detect larger objects, such as hands, that are in proximity of the interactive surface. Achievable distances are up to 30 centimeters and the sensors may be applied below thick non-conductive material.

2.1.3. Measuring modes

A classic work in the field of capacitive proximity sensing that will be referenced occasionally in this work is "Electric Field Imaging" by Joshua Smith [Smi99]. One contribution was the introduction of different measurement modes in capacitive sensing, as shown in Figure 2.4. Transmit mode is using a transmitting electrode that

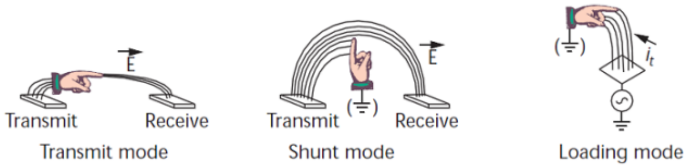

Figure 2.4.: Three measurement modes for capacitive proximity sensing [Smi96]

is coupled to a conductive object; in case of interaction applications typically the human body. The properties of an electric field generated with respect to a receiving electrode will therefore be dependent on the distance of this body, thus extending the achievable range. Shunt mode similarly uses both a receiving and transmitting electrode generating a static field. However, there is no body coupled and any conductive object will ground the field, thus reducing the energy stored, which is measured. This setup is able to work with various transmitters on a single receiver, enabling a higher amount of virtual sensors using limited hardware. The third measurement mode is called loading mode. An oscillating field is induced on a single electrode measuring the capacitance relative to the environment. Any approaching grounded object results in an increased capacitance that is measured periodically.

2.1.4. Materials and geometry

Two major factors that have to be considered when designing an application based on capacitive sensors are the materials and geometry of the electrodes performing the measurements. The material of the electrode should be picked according to the desired application, i.e. if the interaction device has a flexible surface, conductive thread could be used, if it is solid and opaque, the application of solid metal electrodes is viable. Additionally there are other options for transparent materials. While we traditionally associate solid metals to antennas and electrodes this view can no longer be upheld. Transparent conductive layers have been in use for decades now, e.g. in car windows or solar technology. They typically rely on metal oxide layers, polymer layers or in recent years carbon nanotubes [MPLK05]. The most common technology for usage in displays is projected capacitive touch that uses a multi-layer design of insulated ITO electrodes that are able to detect the movement of several objects close to the surface [BO10b]. However, they are typically tuned to allow operation within a small distance of $1cm$ or less. However, they are typically tuned to allow operation within a small distance of $1cm$ or less. One recent work was evaluating different types of electrode materials in terms of their spatial resolution at different distances between object and electrode [GPBB*13], focusing on larger distance proximity measurements. They benchmarked both ITO and PEDOT:PSS. The first is a thin layer of indium-titanium-oxide, a highly conductive metal layer that possesses good optical properties. PEDOT:PSS is a conductive polymer that has a lower conductivity and slightly less appealing optical properties. In conclusion they evaluated that while copper has still the best properties, at least ITO can be considered a suitable alternative in applications that require optical clarity, as shown in the achievable spatial resolution given in Figure 2.5. The most common technology for usage in displays is projected capacitive touch that uses a multi-layer design of insulated ITO electrodes that are able to detect the movement of several objects close to the surface [BO10b]. However, they are typically tuned to allow operation within a small distance of 1cm or less. Another area that is strongly influenced by the intended application is the geometry, whereas the electrode is considered the part of the electronics directly attached to the measurement circuit. This may range from simple straight wires or plate electrodes to complex optimized multidimensional structures specifically designed for a single task. Even though it is aimed at touch or near-proximity sensing we will give a

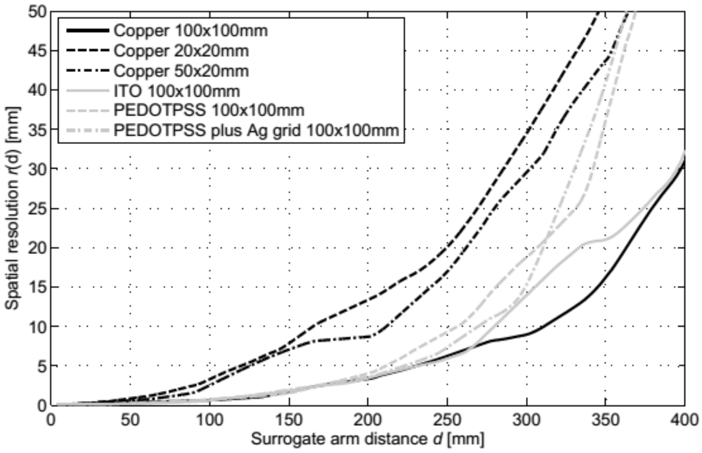

Figure 2.5.: Spatial resolution of different materials at various distances [GPBB*13]

short overview of multi-layer designs for touch screens that have been reviewed by Barrett and Omote [BO10a]. They are designed to measure mutual capacitance, i.e. the resulting capacitive properties between a sending and a receiving electrode that are intersecting. If a sensible excitation and measuring process is used, multiple nearby objects may be reliably detected. A simple example is two layers of perpendicular straight line electrodes - used

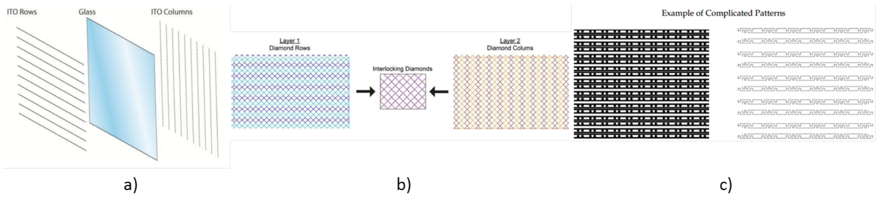

Figure 2.6.: Examples of multilayer layouts for touch screens - grid (a), interlocking diamonds (b) and trademarked complex patterns (c) [BO10a]

by the first iPhone (Figure 2.6 - a). Another example uses an interlocking diamond shape [DL01] to create a good spatial coverage (Figure 2.6 - b). Finally, there are numerous other complex patterns that are often trademarked by the companies that have developed the respective controller. One example is given in (Figure 2.6 - c).

Capacitive proximity sensing applications are typically less concerned about intricate designs, but instead use varying electrode sizes and placement over a larger area. As previously mentioned the purpose of capacitive proximity sensing is the detection of objects and their properties. There are numerous factors that can influence the geometrical layout, but they can be abstracted into the following categories:

- Number of objects
- Object size
- Desired spatial resolution

Going back to our example of touch screens, we have small objects, a higher number of those (usually up to 10) and require a high spatial resolution to select small items on the screen. The result is a fine multilayer grid, using mutual capacitance to simplify multi-object recognition, fine electrode spacing to achieve a high spatial resolution and thin or transparent electrodes to guarantee good optical properties. A similar rationale can be applied to other applications. If we take the smart couch by Große-Puppendahl et al. the aim is to detect the presence and posture of one or more persons on a couch [GePMB11]. This necessitates detecting large body parts such as head, torso or limbs. There is no fine-grained spatial resolution required, allowing a reduction the number of sensors and it was assumed that a maximum of two persons are on the couch. Furthermore the electrodes are placed below the upholstery, thus requiring a reasonable detection distance. The resulting electrode placement can be seen in

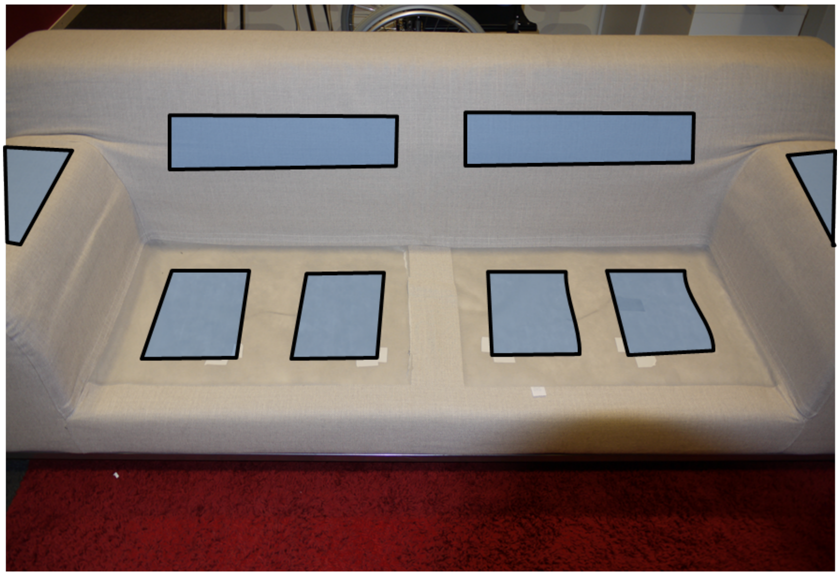

Figure 2.7.: Electrode placement below upholstery [GePMB11]

Figure 2.7. The layout was designed under the additional constriction of using a single sensor kit, supporting up to eight electrodes. Regarding placement it is most important to distinguish two persons and different sitting positions, thus four electrodes are placed below the sitting area. In the back there are two electrodes spread over the entire width to determine the presence of the upper body close to the backrest. The electrodes in the armrests determine a head and are primarily suitable for detecting lying positions. In consequence this setup is suitable for detecting multiple sitting persons, infer information about their sitting position and recognize lying persons. Regarding those postures it showed good results in the prototype's evaluation [GePMB11].

A third and final example for the rationale of electrode placement is the TileTrack system by Valtonen et al, a capacitive person tracking system using floor tiles [VMV09]. It is a transmit mode system that has the transmitting electrodes placed below the floor tiles and the receiving electrodes are placed in the walls of the area. The main goal of the system is the tracking of persons on the surface. Thus the floor area should be mostly covered by electrodes to establish a good transmission link to the bodies. The receiving electrodes should be able to pick up all signals generated by the body. Valtonen et al. picked wire or plate electrodes that went from floor level to a height of 190cm that covers most typical body sizes. While the system has some shortcomings with regard to applicability in larger rooms, the design rationale is appropriate for narrow rooms or when only movement close to walls has to be detected and had a reasonable precision in their evaluation. Looking at the above examples it becomes apparent that the proper selection of materials and geometry is highly depending of the desired application. In consequence it is difficult to give generic guidelines independent of the application. After reviewing the different application domains in the next section we will revisit this topic in section 5.4.

2.1.5. Data processing

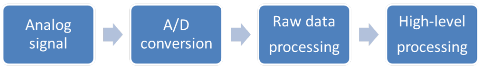

Figure 2.8.: Abstracted sensor data processing pipeline

In order to acquire usable data from any digital sensor an analog signal has to be acquired and processed. A simplified typical processing pipeline for this is shown in Figure 2.8. This basic structure is also applicable to the processing of capacitive proximity sensor data. The analog signal is the capacitance of an electric circuit that can be digitized using different methods, e.g. by using the quantized discharge time of the circuit. In the following section some typical steps of raw data processing and high-level processing for capacitive proximity sensors are presented and discussed.

2.1.5.1. Raw data processing

Raw data processing of capacitive proximity sensor data is primarily intended to compensate for sensor noise and environmental influences. Noise is an inherent property of any measurement system and describes random unwanted data that is added to a signal. Environmental parameters can have strong influence on the signal of a capacitive sensor system. These effecting factors include temperature, humidity, composition of the air, or grounded objects in close proximity. There are numerous additional preprocessing steps that can be taken, such as different multiplexing methods that may be required in some hardware settings, or signal quantization that reduces the outgoing data to a distinct set of values in order to simplify post processing of different applications. These will not be further discussed in the scope of this work.

Noise Reduction In order to deal with noise, some sort of filtering is typically applied. Filtering describes a set of methods that attenuate the parts of a signal that are relevant in a given application. In capacitive proximity sensing we are dealing mostly with high-frequency noise that is added to the signal. Therefore, low-pass filtering can be used to deal with this influence. The most typical examples are average filters that take various samples and calculate an average value, and median filters that are sorting a set of samples and select the median element. Each of those filters has a plethora of potential adaptations that are not too specific to discuss in this limited space. Some adaptations are discussed in the specific prototype sections.

Table 2.1.: Baseline calibrations terms and methods

Name	Description	Application
Initial calibration	First set-up of baseline at system start, e.g. by taking the average over various samples	Required for any application
Static baseline	Baseline that does not change at run-time	For static environments
Dynamic baseline	Baseline that changes over time	For non-static environments
Drift	Change of system response to environmental factors at run-time	-
Drift compensation	Methods to account for occurring drift, by changing the baseline value	Non-static applications
Recalibration	Change of the baseline value at a specific point in time given a set of rules	Non-static applications

Baseline Calibration A very important aspect of capacitive raw data processing is signal calibration. The generated electric field is subject to changes over time, if either intrinsic parameters change or the environment is modified. Some specific examples include the electronic components heating up, the environmental temperature changing, or objects being moved in and out of detection range. Therefore it is essential to have a well-calibrated and adaptive baseline; that is the sensor signal generated in the environment without the presence of any object that we want to detect. Again, there are numerous methods to adapt and configure the baseline. We have collected a few common terms and methods and give some pointers regarding their application. The results are shown in Table 2.1. If a dynamic baseline is used, a set of rules will have to be defined that determines at which points

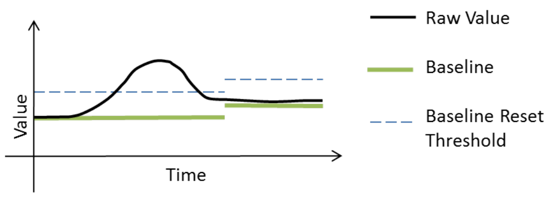

Figure 2.9.: Example of baseline reset using a threshold rule

in time the baseline has to be recalibrated, what specific methods should be used and the set of parameters that control the methods. One simple example is to define a threshold level that triggers a baseline calibration, as shown in Figure 2.9. The raw signal is above the threshold, indicating the presence of a detectable object. Afterwards, it falls back down below the threshold, yet stays for a certain time above the baseline. This triggers a reset of the baseline after a certain amount of time.

2.1.5.2. High-level processing

High-level processing assumes that we already have calibrated (and possibly normalized) sensor values that are used in further steps. The goal of any capacitive sensing application is the acquisition of information about a detectable object, e.g. its current position, the material used or the shape. In order to get this information we need to use knowledge about the object and intrinsic properties of the sensor system. In this section we will discuss methods to combine data from various sensors using the system properties, how to track the position of an object using different methods and how to recognize specific features. An overview of the methods in abridged form is given in Table 2.2.

Table 2.2.: Overview of high-level processing methods for capacitive proximity sensors

Name	Description
Sensor data fusion	Combining sensor data into a shared representational format
Uniform fusion	Sensor data fusion that combines all data into a single common format
Heterogeneous fusion	Sensor data fusion that combines groups of data to serve multiple purposes
Object tracking	Continuous identification of an object within the systems range
Single object tracking	Methods to realize object tracking for a single detectable object
Multiple object tracking	Methods to realize object tracking for multiple objects
Feature recognition	Identifying certain parameters of an object within the system range

Sensor data fusion Sensor data fusion in its most general terms describes “the theory, techniques and tools which are used for combining sensor data, or data derived from sensory data, into a common representational format” [Mit07]. Using the combined information from various capacitive proximity sensors we are able to generate high-level information that exceeds the capabilities of a single sensor. We can distinguish uniform fusion that uses the information from all involved sensors in one common way or heterogeneous fusion that combines groups of involved sensors that serve multiple purposes, yet are attached to a single system. A simple example for the latter would be a single large electrode sensor that detects the presence of a hand from a farther distance and then a combination of various small electrodes that track single fingers. Sensor data fusion often requires taking into account some additional information we possess about the system. A classic example is the precision or bias of the sensor. Various methods, e.g. the class of Kalman filters, use weighted information from several sensor sources [WB95]. If we know how that a certain sensor is only half as precise as another one working in collaborating, the weighting factors can be adapted accordingly.

One of the most important additional information we use when fusing data of capacitive proximity sensors, is the geometric layout of the system. This describes position and size of all electrodes that are integrated. Using this information is crucial when trying to localize an object. A simple example would be applying a weighted average algorithm on a set of sensors. In order to determine object location relative to the plane a weighted average algorithm is used. The linear object location $\overline{x}$ is calculated using the sums over sensor positions x_i and sensor values v_i as weight:

$$\bar{x} = \frac{\sum_{i=1}^{n} v_i x_i}{\sum_{i=1}^{n} v_i} \tag{2.6}$$

Using similar methods we are able to determine the location of multiple objects or additional dimensions of the position. However, it is possible to use other information in the fusion process as well. The electrode material may result in a different response and thus should be treated differently in a fused data representation and can be weighted. Another example is the shape of the electrode that may result in different responses. How to apply sensor data fusion is strongly depending on the application and the desired common representation that is most suitable for subsequent calculations.

Object tracking In the previous section about sensor data fusion we have shortly discussed a method to determine the linear position of a single object using a linear array of capacitive proximity sensor. This is a basic example of a group of methods associated to object tracking. In computer vision applications they can be defined as "the problem of estimating a trajectory of an object in an image plane as it moves around a scene" [YJS06]. The analogy to capacitive applications is viable if we consider a 3D scene and a distinct interaction space instead of a scene. Capacitive proximity sensors allow the detection of conductive objects within their range. However, as this presence is determined indirectly using the influence on an electric field it is not possible to get a direct association between the actual distance between sensor and object and the resulting sensor value. The created electric field is only analytically descriptive for very specific, theoretic classes of objects [Bax96]. Nonetheless, we are able to get a relative distance measurement. If we combine this proximity value using geometric information about the electrode location we can infer the relative position of an object in the sensing area. The weighted average method presented in the previous section is one option for relative positioning. Another method is trilateration, similar to many radio-based localization applications, that uses the known location of three or more points and the known distance to the position to be determined. In case of capacitive proximity sensing this position is determined relative to the electrodes as there is no absolute distance measurement. A more complex example for direct calculation was presented by Smith, who formulated the issue of detecting multiple objects as a forward problem and used numerical methods to estimate the position and orientation of two hands [Smi99]. A

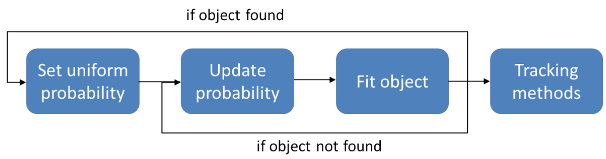

Figure 2.10.: Generic pipeline of probability based methods of capacitive proximity sensing

second class of methods to track objects is not relying on direct geometrical calculations but instead formulates a numerical solution to a probability distribution. The initial assumption is that the probability of an object to be at a certain point in the detection area is uniform. The methods then follow a few basic steps, as shown in Figure 2.10. At first the probability is updated based on the current sensor readings and a priori knowledge that we have about the system. Afterwards we try to fit the objects into the resulting probabilities. This step may or may not work, meaning that it may result in no object found. In the latter case the process will have to start at the beginning. If an object is found the probability update may use the current object location in the update algorithm, thus starting with a non-uniform probability distribution. One example for probability-based object recognition using capacitive proximity sensors was presented by Grosse-Puppendahl et al. [GPBKK13]. Using a model suggested by Smith the basic idea is using the assumption that an object may be present anywhere, remove regions where no objects can be present and then fit an object into the remaining space. This method additionally uses particle filters to track object locations over time. This also allows tracking multiple objects. Throughout the years various methods have been suggested for supporting multi-object tracking using capacitive sensors. Touch screens often use inversion of the sender signal to reliably detect the positions of multiple points; however, this method can't be used in proximity applications [WF07]. Some of the previously presented methods support the tracking of two or more objects. There are still various limitations, particularly if not only the object location but also various other features such as rotation should be tracked. This is still an area of ongoing research, leading to the next area of high-level processing - feature recognition.

Table 2.3.: Feature recognition methods

Name	Description
Data-driven methods	Directly associate input data to output features using various methods, e.g. machine learning and training data
Model-driven methods	Input data is manipulating a pre-defined model of the system that is latter mapped to the output
Neural networks	Computational models using a network of neuron-like objects that are often used in machine learning
Pattern recognition	Methods that look for certain patterns in a set of input data
Semantic mapping	Methods to realize object tracking for a single detectable object

Feature recognition Feature recognition is primarily used as a term in image processing, traditionally in computer-aided design applications to recognize specific geometric properties of an object but also picture analysis, e.g. in facial recognition [HPR00, BHK97]. In the domain of capacitive proximity sensing, feature recognition can be defined as the acquisition of non-location information from any detectable object. An important feature in industrial applications is the material of an object [Bax96]. With regards to recognizing additional features a system was presented by Wimmer et al. - Thracker [WHKS06], a prototype augmenting a regular monitor with capacitive proximity sensors. In addition to recognizing hand position the system is able to detect grasp gestures, which can be used to select items on the screen and perform pick and drop operations. Capacitive sensors can also be used to distinguish between persons and a children's seat on the passenger side of a car [GZBB09]. The methods to recognize the features can be divers, ranging from typical machine learning algorithms, to model-based approaches. An incomplete list is given in Table 2.3. In order to keep this work contained we refrain from a deeper discussion at this point.

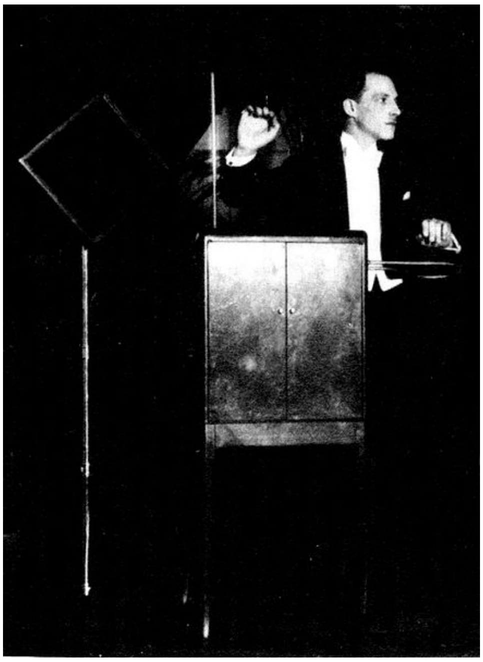

Figure 2.11.: Leon Theremin playing his epnoymous electronic musical instrument [Gli00]

2.2. Capacitive proximity sensing applications

2.2.1. A brief history of capacitive proximity sensing

In the last decades of the 19th and the beginning of the 20th century a considerable number of inventors and scientists performed research on the application of electric systems, sparking innovations such as electric lighting, electric motors, telegraphy and radio communication. Lev Sergeyevich Termen or Léon Theremin in the American naming was a Russian inventor most famous for designing the theremin. This early electronic musical instrument could be played without touch. One hand is controlling the pitch and the other the volume by changing the distance to an antenna. Initially designed as a motion detector, this device is transferring the influence of the human body on an oscillating electric field to an audible sound [Gli00].

Electric field imaging was a research focus at the MIT in the 1990s. A research group in the Media Lab division including Joseph A. Paradiso, Thomas G. Zimmerman, Joshua R. Smith designed various sensing devices and evaluated various applications in HCI [ZSP*95] [Smi99]. - NEC passenger seat - Paradiso/Smith - Wimmer - Touché - Swallowing - Hamburg Gruppe - Finnland Anwendungen

2.3. Sensor systems in smart environments

In the most general definition a sensor is a device that transforms a physical property into an observable signal. This definition includes traditional systems such as mercury-based thermometers or hair-based hygrometers. Yet nowadays we are usually considering digital sensors that transfer the measured property to a binary signal that can be further processed by computing devices. A common variety is the smart sensor that provides additional functionality beyond generating a correct sensing signal [Fra13]. The main goal is to simplify installation and maintenance of distributed sensing systems by having processing close to the measurement device. Early considerations in this domain were put to the standard family IEEE 1451 - IEEE Standard for a Smart Transducer Interface for Sensors and Actuators between 1997 and 2007 [iee07]. An additional concept is the Virtual Sensor that includes digital signal processing and conditioning and therefor abstracts the processing steps from devices interfacing the sensor. The number of available sensors is very high, but it is possible to restrict them based on application domain. Lewis and Cook et al. [Lew04, CD07] have proposed a collection for smart environments focused on wireless sensor networks. The overview is shown in table 2.4. This sensor categorization is based on

Table 2.4.: Sensors for smart environments [CD07]

Properties	**Measurand**
Physical properties	Pressure, temperature, humidity, flow
Motion properties	Position, velocity, angular velocity, acceleration
Contact properties	Strain, force, torque, slip, vibration
Presence	Tactile/contact, proximity, distance/range, motion
Biochemical	Biochemical agents
Identification	Personal features, RFID or personal ID

the property to be measured and is agnostic to the specific measurement technology. Physical properties, such as pressure, temperature, humidity and flow, can also be noted as environmental properties. They are measurements that determine the state of the smart environment, e.g. temperature in different rooms, or the current water usage. Motion properties denote the movement parameters of actors in this environment and can refer to both humans and machines. Angular velocity is important in self-localization of robots in an environment. Contact properties groups the different types of interaction between surfaces in the smart environment and actors. Presence as a group is similar to motion paramteres, but does not require a series of measurements for tracking an actor. Biochemical sensors enable measuring the presence of specifc chemical compounds in the environment and are most suited for measuring pollution or air quality. Finally, identification of actors allows to provide personalized services and can be realized with different methods ranging from tags to biometric systems.

While this listing provides a decent overview of sensing properties in smart environments it is abstracted from sensor technologies. Various types of sensors, including capacitive proximity sensors, allow us to detect multiple of these properties and thus providing a higher flexibility. Therefore it is possible to provide an inverse listing of sensor technologies that allow measuring different properties. A short overview of sensor technologies with this capabilities and that are commonly used in smart environments is given in table 2.5. In the following sections I want to give an overview on how these sensor systems are used in this domain, in order to provide a basis for the benchmarking model that will be introduced in section 3.

2.3.1. RGB cameras

A RGB camera is an image processing device that processes light in the visible spectrum, similar to the human eye. Modeled after the retina it has three distinct color channels - red, green and blue. There are different methods available to distinguish these channels from visible light, such as Bayer filters (Figure 2.12) in front of a single

Table 2.5.: Sensing technologies and measured properties

Technology	Properties
RGB cameras	Motion, Presence, Identification
Infrared cameras	Motion, Presence, Contact
Ultrasound sensing	Motion, Presence, Contact, Identification
Microphone arrays	Motion, Presence, Contact, Identification
Radiofrequency sensing	Motion, Presence, Identification

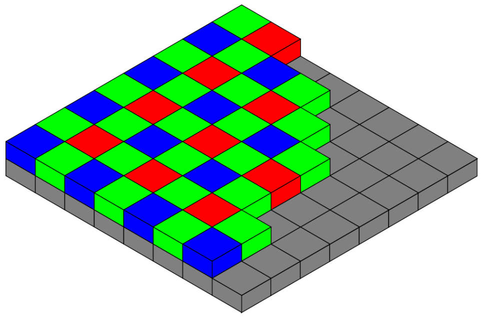

Figure 2.12.: A bayer pattern on a sensor in isometric perspective [Com06]

sensor or using multiple sensors behind a prism. The usage of cameras in smart environments is very common. I will present five different examples and afterwards will specify how they are linked to the properties that were defined previously. Tabar et al. have been using a combined system of cameras, RF transmitters and wearable sensors in a home care scenario [TKA06]. The cameras are used to improve the accuracy of the accelerometer-based fall detection by eliminating false positives. Once a fall event occurs an algorithm tracks the posture of detected humans in the scene (Figure 2.13 - left). They used an edge detector to distinguish the human body from other objects and applied a heuristic to differentiate lying and standing. Additionally a face detector was used to improve the recognition of human objects. Combining this with information from the fall detecting sensor and a RF based localization system they were able to achieve a good reliability in eliminating false positive alerts.

Pentland and Choudhury provided an overview of vision-based face recognition systems in the domain of smart environments [PC00]. The systems are able to identify users and recognize facial expressions. The proposed applications in smart environments include personalized shopping experiences based on customer recognition, behavior monitoring in child care facilities and emotion-aware systems that react to the user's current awareness. The described techniques include PCA-supported, eigenvector-based classification, face-based localization and systems based on local feature analysis (Figure 2.13 - right). Newer systems are able to operate well in unconstrained environments, that include varying expression and illumination, ageing of persons, occlusion and disguise [WYG*09].

An example for a novel image processing method that is useful in smart environments was presented by Vu et al. in 2012 [WRS*12]. They are using temporal variances of pixel values to exaggerate spatial movements and color changes that would typically be invisible to the naked eye. The method is called Eulerian Video Magnification and uses a combination of spatial decomposition and temporal filtering applied to adjacent frames.

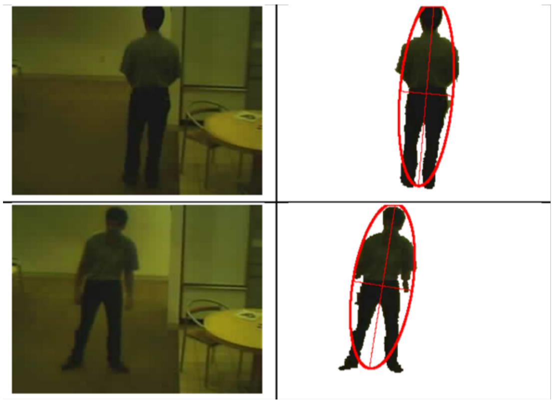

Figure 2.13.: *Left*: Tracking of body masks using cameras [TKA06]. *Right*: Eigenfaces created from input picture set [PC00]

It can be tuned to different time-frequency bands to attenuate different classes of signals. Some of the proposed applications include the tracking of breathing rates of infants by attenuating chest movement, or the tracking of subtle movements, such as vibration in appliances. The example shown in Figure 2.14 on the left is using a magnification of colors, in order to identify the heart rate of a person. The latter can be used for personal health applications, e.g. by integrating the system into the bath room mirror to provide an unobtrusive daily measurement and give the user feedback over a longer period of time.

A final example in this section is the FoodBoard, a smart chopping board that uses image processing to recognize the food items that are put on it [PJS*13]. It is shown in Figure 2.14 on the right. To enable a thin footprint, ambient light is transferred to a camera using glass fibers. The picture is reconstructed and segmented, allowing to identify different items of food that are placed on it. The classification is based on a combination of Fast-Hessian and color histogram feature extractors. Pham et al. were able to distinguish 12 different ingredients with an accuracy between 59% and 93%. The system can be used to support dietary monitoring, give recipe guidance or support visually impaired users in identifying and tracking food.

2.3.2. Infrared cameras

Infrared imaging is using the same sensors that are suitable for visible light imaging. The difference is that they are tuned to collect electromagnetic waves of a lower wavelength that are just outside of the visible spectrum. This allows for distinct applications, such as thermal imaging, as it is possible to detect heat radiation. One of the earliest prototypes in Ubiquitous Computing designed by PARC was using the ORL Active Badge, an infrared emitter developed by Olivietti Research Laboratories that was used to identify persons operating in the environment [Wei91] (Figure 2.15 Top Left). In smart environments the most common application is using infrared cameras in combination with infrared light sources. This allows to illuminate spaces without visible artifacts to the user, thus providing imaging capabilities in dark rooms, or very specific conditions that may be required by a certain application. Another very interesting option is to use a specific projection of patterns into the scene. Analyzing the returning infrared light it is possible to infer the depth of specific pixels of the camera (Figure 2.15 Top Right). This variety is called a depth camera. Particularly in the last few years the research in this domain has expanded strongly, sparked by the availability of an affordable depth camera/RGB camera combination - the Kinect by Microsoft [Zha12]. On the following pages we will present various examples of

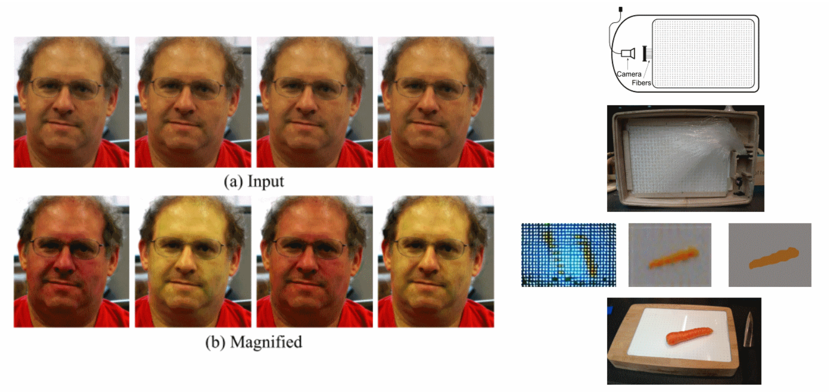

Figure 2.14.: *Left*: Eulerian Video Magnification to attenuate the human pulse with original (a) and amplified (b) video sequence [WRS*12]. *Right*: FoodBoard schematics (top), underside view (second row), original, reconstructed and segmented image (third row) and final system (bottom) [PJS*13]

how this device can be used in smart environments to enable different applications in interaction and activity tracking.

Sung et al. have presented a system that is tracking activities of daily life based on the movements of a skeleton that is provided by the Kinect API [SPSS11]. This skeleton model is based on a pose reconstruction algorithm developed by Shotton et al. [SFC*13] and is used in many different Kinect-based applications. The algorithm is using a method called hierarchical maximum entropy Markov model (MEMM). Each activity is considered to be composed of sub-activities. Based on this assumption a two-level graph is generated using dynamic programming. The system was tested with twelve activities performed in an office, a kitchen, a living room, a bathroom and a bedroom. If the person was part of the training set the precision was 84.3% and 64.2% for unknown persons. Some example activities of the acquired dataset are shown in Figure .

A novel method to use the Kinect for fast 3D reconstruction of scenes was presented by Izadi et al. in 2011 [IKH11] (Figure 2.15 Bottom Left). The basic premise is to use a fast registration method to combine point clouds that are generated by the system and continuously extend and optimize the current model of the scene. They are using a GPU-based ICP implementation to track the position of the camera in six degrees-of-freedom. This allows to reliably integrate the different point clouds into a single voxel grid that can be used to represent and render the scene. They proposed a number of applications ranging from phyisical simulation of particles in the scene, to system control based on segmenting and tracking the user's hands and their interaction with arbitrary surfaces in the environment. Figure shows some results of the touch recognition in a reconstructed scene.

Galen has presented a method to use a Kinect in a kitchen to provide touch free interaction [Pan12]. Two different interaction schemes are proposed. The first allows to control the applications with messy hands, the second enables to use other limbs if the hands are currently occupied (Figure 2.15 Bottom Right). Three different test applications have been implemented. A recipe navigator that allows to open and navigate through different recipes. A timer that enables setting different alarm times, similar to a kitchen clock and a music player that can be controlled to choose different stations according to preference. A real life test with five subjects did also test installation complexity, which was deemed favorable. There were some concerns in clearly distinguishing commands from random movements performed in the kitchen.

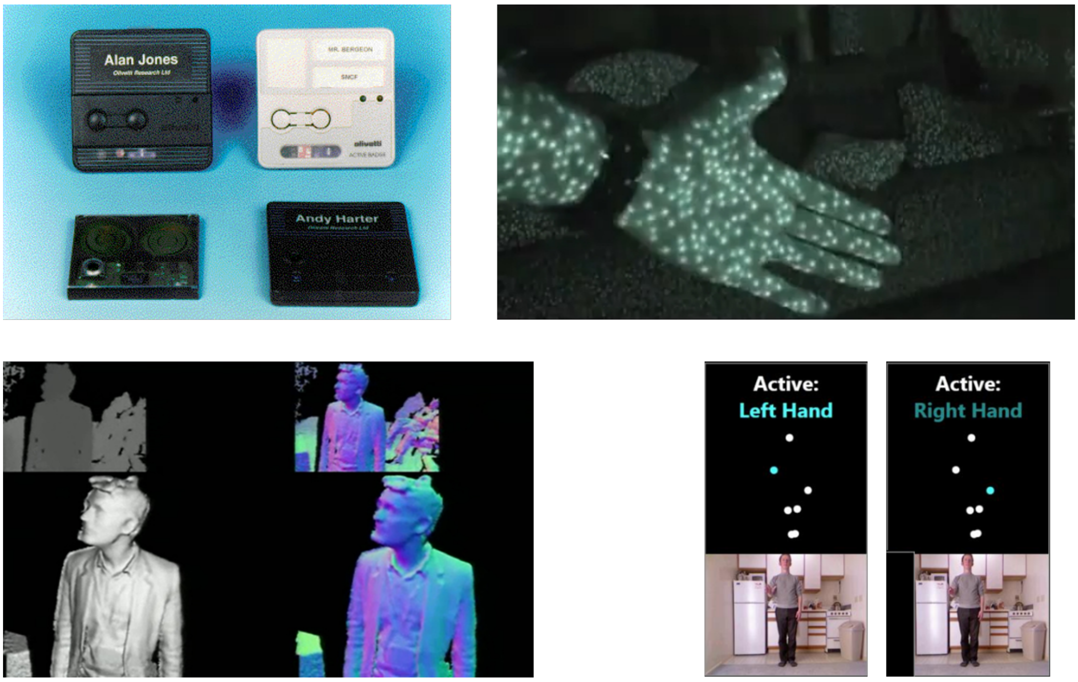

Figure 2.15.: *Top Left*: ORL Active Badge [Wei91]. *Top Right*: Kinect infrared projection [Zha12]. *Bottom Left*: Kinect Fusion reconstrucion [IKH11]. *Bottom Right*: Kinect kitchen interaction [Pan12]

The final system based on infrared cameras is an immersive telepresence system developed by Beck et al. [BKKF13]. Telepresence enables persons to be present as a representation in a remote place. A typical example is video conferences, but advanced system may include robotics or virtual elements. The presented system is using a single Kinect for each participant. A 3D representation is created based on the depth information of the infrared camera and on-the-fly texturing using the included RGB camera. The virtual user representations are put into a shared virutal environment and can interact with each other. Some variations were tested where local and remote users were decoupled, side-by-side or face-to-face. Different tracing and pointing gestures are supported. The supported applications included the exploration of a virtual city.

2.3.3. Ultrasound sensors

Ultrasound sensors allow detecting sound wave signals that have a frequency beyond 20kHz and are thus not audible to humans. Their propagation and reflection properties are similar to audible sound waves, thus the generated measurements can be similar. While there are natural sources of ultrasound waves the applications in smart environments do rely on active systems, that combine sound generators and sensors that measure the resulting signal. By timing the time distance between sending the signal and receiving a response it is possible to measure distances between the sender and different object. If various receivers are used it is possible to localize the sound source, making ultrasound sensing a popular candidate in indoor localization systems. In Figure we can see a sketch of the basic functionality of ultrasound sensing systems on the left, and an example of localization

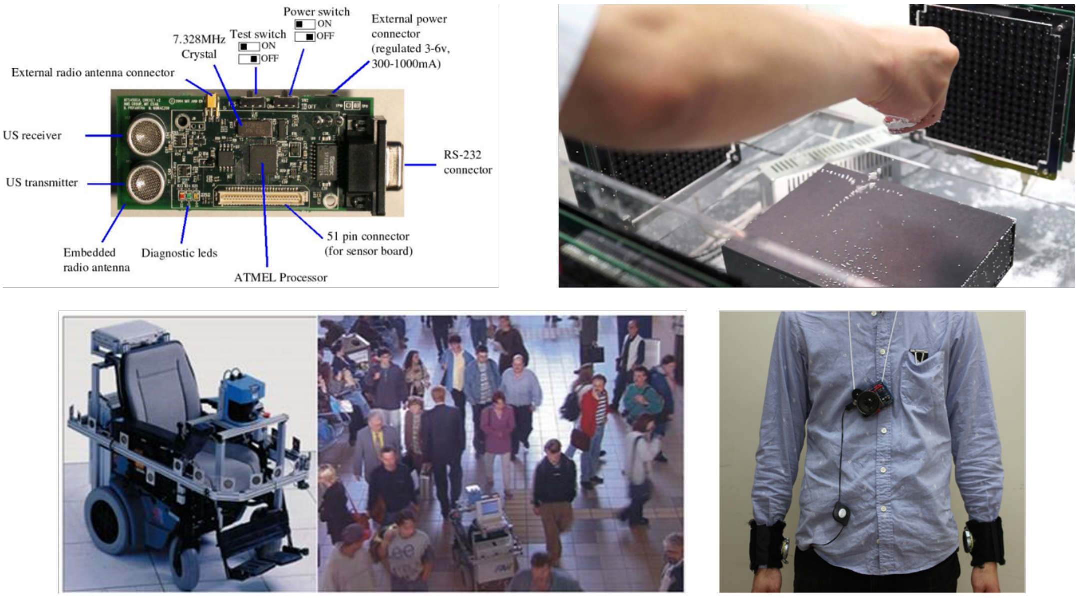

Figure 2.16.: *Top Left*: Cricket Indoor Localization hardware [PCB00]. *Top Right*: Mid-air particle manipulation [OHR13]. *Bottom Left*: Robotic wheelchair MAid with ultrasound range sensors [PSF01]. *Bottom Right*: Activity and context recognition using ultrasound sensors [WTT13]

using three receivers and a single source. We will present four different examples on how ultrasound sensors are used in smart environment applications.

The Cricket developed by Pryantha et al. is an example for an indoor localization system based on a badge the tracked object has to wear [PCB00]. The badge is periodically sending signals to a set of beacons that determine the distance and calculate a location. Initially it was supposed to solely rely on radiofrequency signals, but was modified to use a combination of RF and ultrasound. The rationale of this decision is the considerably slower speed of sound that simplifies measuring the time-of-flight and thus allows for more precise distance measurements. Consequently this also leads to a better precision in the localization algorithm. Additionally, the system uses some novel methods to deal with interference and multipath issues, that is dealing with reflected signals. As potential applications they propose space-dependent services that are provided as the user is identified in a certain region and guidance scenarios on a floor plan. The hardware is shown in Figure 2.16 (Top Left).

The robotic wheelchair MAid (Mobility Aid for Elderly and Disabled People) was designed to support and transport people with limited motion skills [PSF01]. It is based on a commercial wheelchair that has been equipped with an intelligent control and navigation system. The system includes an ultrasound-based range finder that allows to detect obstacles in the path and circumnavigate around (Figure 2.16 Bottom Left).

Watanabe et al. investigate the role of ultrasound sensors in recognizing activities and gestures of a user [WTT13]. The system is comprised of a microphone attached to a necklace and two speakers that are attached to each wrist. Based on the acquired volume and evaluation of the Doppler effect it is possible to determine both distance of the wrists from the neck and the speed of movement (Figure 2.16 Bottom Right). Watanabe et al. want to determine if this system allows for similar performance compared to other body-worn systems based on accelerometer and gyroscope data. Additionally it was evaluated if external microphones can perform as well as the neck worn microphone. The system was able to recognize 87% of activities and gestures in a set of 10 test persons if no influencing sound was present. The ultrasound did also improve results if environmental noise was present.

A recent project at the University of Tokyo evaluates the potential of ultrasound in manipulating small particles in free-air [OHR13]. Using standing waves to create sound pressure notes it is possible to apply a force to small particles that is sufficient to counteract gravity. Using a set of ultrasonic phased arrays it is possible to create these sound pressure nodes at arbitrary positions in three dimensions (Figure 2.16 Top Right). Ochiai et al. use this to move small objects around and investigate required object properties and their floating properties. So far the moved items have to be very light and smaller than 2 mm in diameter, resulting in the usage of polystyrene. The technology is also able to hold and move small amounts of fluid. Suggested applications include object manipulation in microgravity environments and projected haptic feedback systems.

2.3.4. Microphone arrays

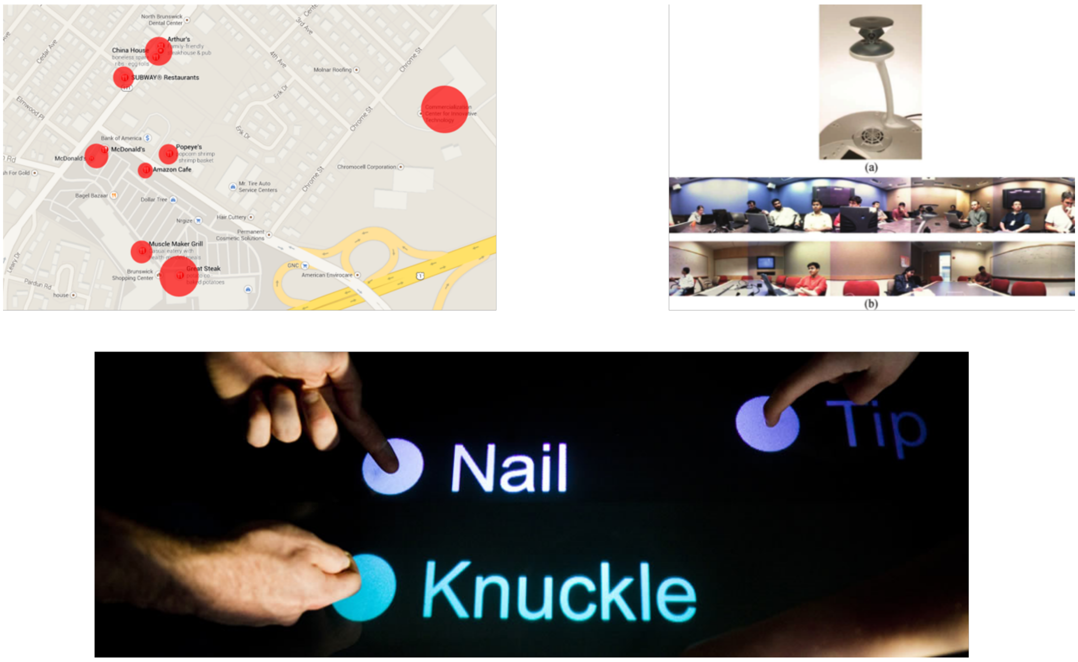

Figure 2.17.: *Top Left*: Visualization of speaker count in different areas [XLL*13]. *Top Right*: Directional microphone and conference room for speech source localization [ZFBZ08]. *Bottom*: TapSense detecting different tap events [HSH11].

Microphones are signal receivers that are tuned to detect sound frequency ranges that are audible by humans. Typically they consist of a piezo element that transfers vibration to an electric current that is amplified. Most human activities produce some kind of sound. As opposed to the presented ultrasound system, microphones typically operate passive. That is there is usually no dedicated signal source, but instead naturally occurring sounds are picked up. By combining microphones or arrays thereof with data processing systems that are aimed at analyzing specific sound patterns we are able to get feedback on human activities. Looking at smart environments there are numerous use cases that can benefit from microphones as sensors. In this section we will present four different systems that cover a large variety of different applications, ranging from breathing rate detection to estimating the number of speakers in large groups.

Detecting the breathing pattern of a person has several applications in smart environments. Apart from medical applications that require detecting abnormal breathing in risk groups it is also possible to track training progress using such a system or provide a measure for the current attention level in affective computing. Corbishley et al. investigated using very small microphones in mobile devices to enable detecting the breathing rate [CRV08]. The algorithm is designed to be applicable on single ICs, allowing for miniaturization and energy efficiency. Even with the presence of noise the combined score for true negatives and true positives was as high as 91.3%. Using small and energy efficient systems also enables unobtrusive applications in non-mobile environments, e.g. placing such a system close to the bed to detect the breathing rate while the user is sleeping.

Collaborative applications are an important aspect of smart environments, e.g. to link together meeting places at different places, similar to the presented telepresence applications. If a multitude of speakers is present it gets increasingly difficult to provide a system that enables proper speech transmission for all participants. Using an array of microphones it is possible to focus the attention on the person currently speaking and filter out environmental noise. A project at Microsoft Research investigated using a maximum likelihood of two known techniques, beamforming and speech source localization, to enable a reliable speaker selection [ZFBZ08] (Figure 2.17 - Top Right). Additionally the framework enables a good adaptation even if directional microphones are used that are placed close to each other. The method provides a real life accuracy of more than 90%.

A fairly new work was performed by Xu et al. [XLL*13], called Crowd++. Their idea is to use smartphone microphones to identify the number of speakers in crowded environments. Such system could be used to estimate the number of persons in a given place and potentially react quickly if a crowd panic may occur. The method is based on an unsupervised machine learning classification of short audio parts that are picked up by the individual microphones in the handsets. It was tested by 120 participants in 10 different environments and allowed detecting the number of persons with an average error of 1.5 persons (Figure 2.17 - Top Left). No dedicated hardware is required to achieve this precision, enabling an application using off-the-shelf smartphones.

Microphones can also be used to analyze the mechanical surface waves that occur when objects interact with each other. Harrison et al. have designed TapSense, a microphone based sensor system that improves touch interaction by classifying the sounds created by different objects hitting the surface [HSH11]. In particular different parts of the hand can be distinguished, including nail, knuckle or tip. Potential applications are improving touch interaction on touch screens by enabling different forms of interaction, but also can be adapted to mobile devices, that typically have less interaction space and increase the expressiveness of different touch gestures. The achievable accuracy was between 95% using four different input types up to 99% when using just two input sets, such as finger and pen (Figure 2.17 - Bottom).

2.3.5. Radiofrequency sensing

Radiofrequency sensing is a traditional field for sensors. Radar is a system that uses radio waves to acquire direction, speed, distance or altitude of objects and was developed at the beginning of World War II. This variety is using active sensing and emits radio waves that are reflected by objects. Most applications in smart environments similarly rely on active systems. A popular signal source are WiFi signals, intended to wirelessly transmit information between several systems. The systems are wide-spread, with all smartphones possesing two or more wireless technologies (typically 4G/3G/GSM for long range communication, WLAN for medium range communication and Bluetooth/NFC for short range communication) that can be used in coordination with stations placed in the environment. The signal processing of the wireless LAN sets generates some additional data that

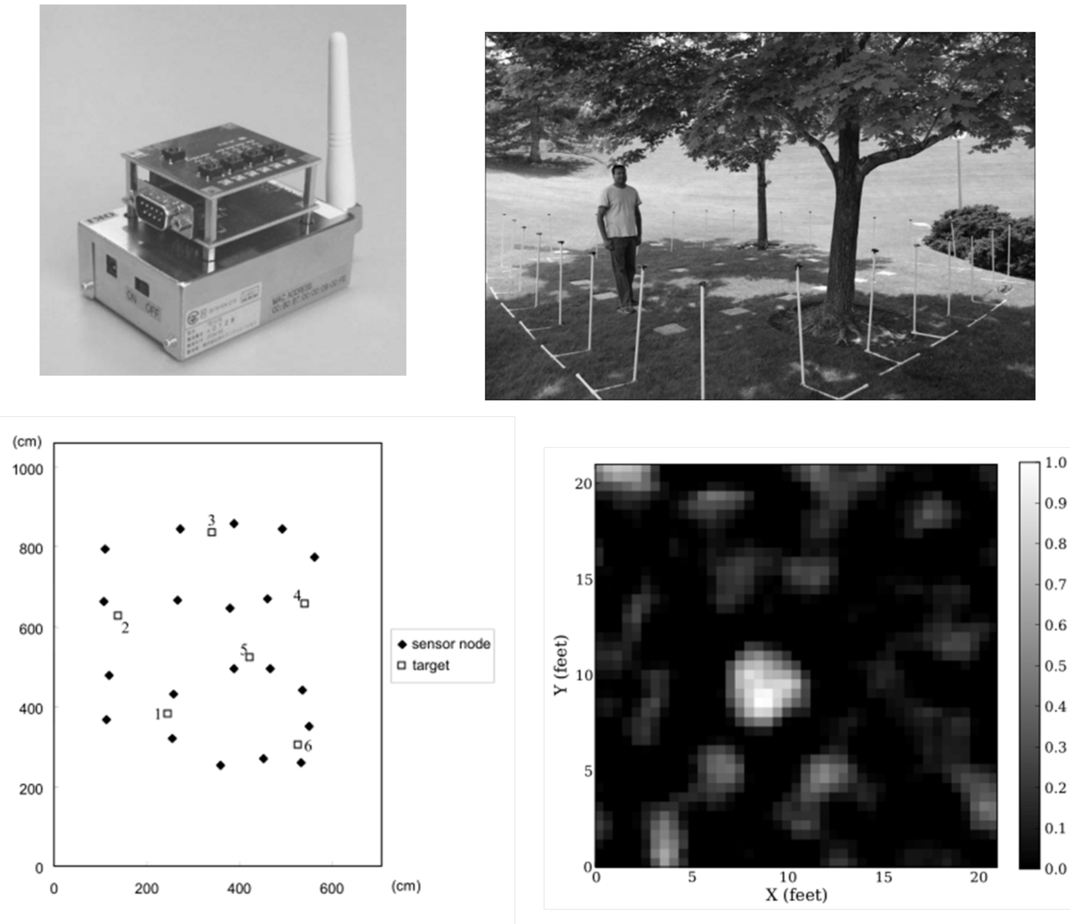

Figure 2.18.: *Top left* ZigBee node. *Bottom Left* Sensors and targets in larger room [SKOM06]. *Top right* Photo of radio tomography setup. *Bottom right* Result attenuated signal [WP10].

can be used, most notably the signal strength (RSSI). We will present different systems that show some different ways how radiofrequency sensing can be used in smart environments.

Radiofrequency based systems are very popular for indoor localization applications. We previously glimpsed at the difficulty of time-of-flight systems in the electromagnetic specturm. Thus most systems rely on a different approach, using the received signal strength (RSSI). If the initial signal strength is known and we have a good estimate how the signal propagates we can estimate the loss of signal strength at a certain distance. An accepted work that helped shaping this domain is the system created by Sugano et al. in 2006 that uses a ZigBee-based network with a limited number of nodes receiving RSSI information [SKOM06]. Based on this it is possible to locate one or more users with an error between 1.5-2m. This often is sufficient to distinguish the room a person is currently in, enabling a room-based system adaptation, which is suitable for many applications. A photo of a ZigBee node and results in a larger conference room are shown in Figure 2.18 on the left.

A different approach for RSSI based systems was presented by Wilson and Patwari in 2010 [WP10]. They are using tomography methods to determine the location of users. By placing a large number of sensors on the outer limits of the environment and creating a unique link between each, it is possible to infer the position based on the signal attenuation when a person moves in the environment. The human body absorbs some of the signal, resulting in a reduced received RSSI in the affected nodes. The error for standing persons was between 0.64cm for a single person and 1.10cm for two persons. An image of the test area and the resulting reconstructed attenuation map are shown in Figure 2.18 on the right.

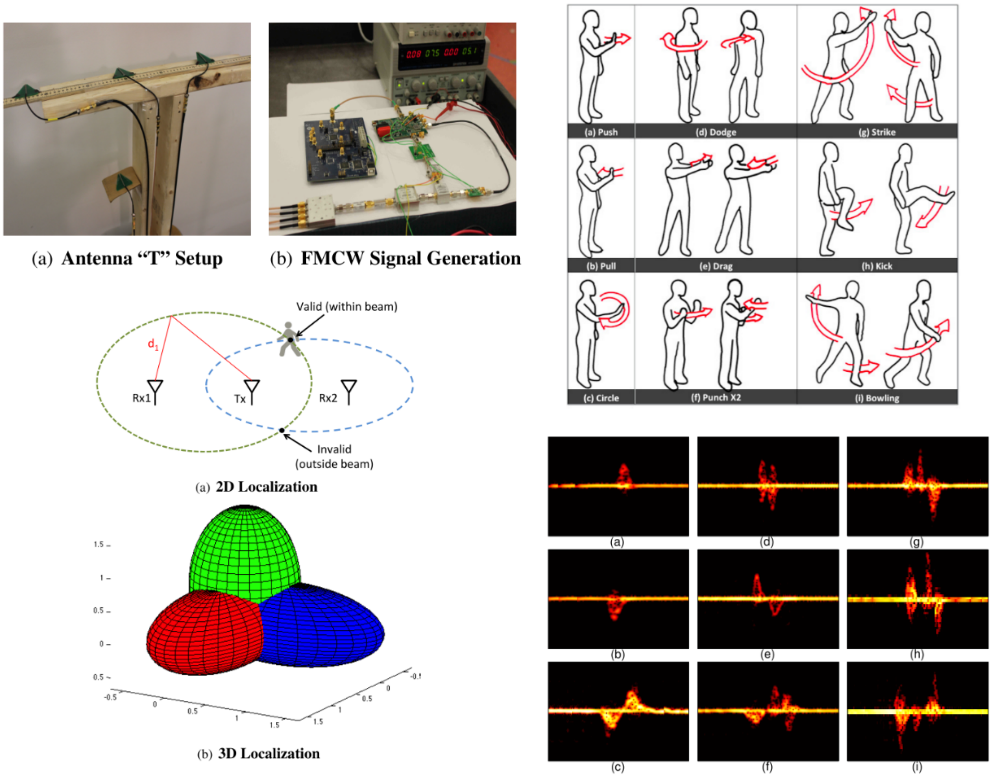

Figure 2.19.: *Top left* WiTrack antennas and signal generator. *Bottom Left* WiTrack 2D and 3D localization method [AKKM13]. *Top right* WiSee supported gestures. *Bottom right* WiSee Doppler profiles of gestures [PGGP13].

A final system that provides a localization is WiTrack, presented by Adib et al in 2013 [AKKM13]. It uses the signal reflected by the human body to provide a location estimate based on time-of-flight. As mentioned previously this is challenging due to the propagation speed within the electromagnetic field. To overcome this issue they are using a method called frequency modulated carrier wave that transfers time differences to the frequency domain. Looking at the spectrum of received signal these shifts can be analyzed. The resulting location has an accuracy of 10-13 cm in x and y and 21 cm in z dimension. Additional use cases are provided in determining coarse arm or foot gestures and enabling an accurate detection of falls of a person. Some limitations of this approach include a restriction to just one person and that the tracked object has to move. In Figure 2.19

on the left we can see pictures of the antenna setup, the signal generator and how the location is determined in two and three dimensions.

The final radiofrequency system I want to present is WiSee, developed at the University of Washington by Pu et al. [PGGP13]. They are using Wi-Fi to enable gesture recognition in large areas without requiring a line-of-sight. They are analyzing the Doppler shift resulting from human activity in order to classify different gestures. They are using MIMO (multiple input multiple output) to distinguish between non-interacting persons in the environment and the person performing gestures. To equip a whole appartment only a single receiver and two transmitters are necessary. The achievable accuracy varies according to persons present and number/type of devices involved but peaks at an accuracy of 94% for detecting nine different whole body gestures. The supported gestures and their Doppler profiles are shown in Figure 2.19 on the right.

2.4. Applications in smart environments

2.4.1. Location-aware services

2.4.2. Natural interfaces

2.4.3. Health monitoring

3. Benchmarking model for sensors in smart environments

When designing a new application or system for a specific purpose, the parties involved have to make a number of decisions regarding the different components, processes and methods that are to be used. Benchmarking is a method mostly used in business practice to compare the performance of processes, products and market entities against one another. A single or a set of different indicators are used to act as metric or calculate an overarching metric of performance that can be compared to other entities [CC89]. This tool is widely used for supporting decisions in different domains. Looking at smart environments, a common challenge is to select a specific sensor technology for any given application. While the majority of systems are following a structured approach in the design process, e.g. by ranking available systems or performing an iterative trial & error routine, so far there has been no generic model that would allow to evaluate the expected performance of a system based on a specific sensor technology. Using a set of common features and an adaptive weighting model it is possible to cover a high number of different applications in a specific domain and thus support the decision process at an early stage of the application design.

In this chapter I will introduce a formal benchmarking model that allows estimating the performance of applications in smart environments based on a specific sensor technology. To build a base for this modeling it is necessary to look into previous works related to technology benchmarking and the selection of specific metrics. Additionally, it is necessary to find a common set of features relevant for smart environments that can be applied to a number of different sensor technologies. The method is evaluated by benchmarking several example technologies and applications based on querying large scientific databases. Finally, the benchmarking method is used to determine different applications suitable for capacitive proximity sensors.

Benchmarking is a tool that is widely used in computing technology [LC85]. Hardware benchmarks compare the performance of different single systems, often seen for GPUs or CPUs to evaluate both theoretical and real-life performance. Some metrics that are used for theoretical comparison in CPUs are FLOPS (floating point operations per second), e.g. measured by Linpack [DLP03], or MIPS (million instructions per second), e.g. measured by Dhrystone [Wei84]. Regarding GPUs the benchmarks include Texel rate (how many triangles can be processed per second) and Pixel rate (how many pixels are processed per second. Real-life benchmarks for CPUs typically included timing specific tasks on applications that are demanding for certain aspects of the CPU, such as video processing, image processing or audio encoding. For GPUs many PC games provide benchmarking tools that allow evaluating the real-life performance of different graphics cards at different settings, e.g. resolution or detail level. The typical metric here are FPS (frames per second) that denote how often the screen content can be rendered in a second. System benchmarks are a step up from single component benchmarks and combine the performance measurements of various components in different scenarios to evaluate the estimated behavior in numerous real-life situations. There are several standardized test suites that provide this functionality, such as SPEC [Hen00]. A common single index that is available for all newer Windows machines (Vista and beyond) is the Windows System Assessment tool that calculates the WEI (Windows Experience Index), a combined score of CPU performance, 2D and 3D graphics performance, memory performance and disk performance [Mic]. The lowest score of all single metrics is chosen to determine a lower bound for expected real-life performance. Ranganathan et al. introduced benchmarking methods and a set for pervasive computing systems [RAMB*05]. They distinguish system metrics, configurability and programmability metrics and human usability metrics. Finally, if different systems of the same category are compared, technology reviewers often use a single index that is calculated based on various aspects of the system. Smith introduced different potential combined metrics that can be used for this purpose [Smi88]. Three different approaches are mentioned, geometric mean, arithmetic mean and harmonic mean. Additionally varieties with a specific weighting are mentioned. Another example for benchmarking whole systems is the EvAAL competition that aims at evaluating different technologies that are applicable in Ambient Assisted Living [CK12]. There are various tracks, including indoor localization and activity recognition. Apart from technical metrics, such as precision, a focus of this competition is on a more holistic approach and thus includes metrics like installation time, user acceptance and interoperability of the solution. Finally there has been considerable work in the domain of identifying suitable metrics for a given benchmark. Crolotte argued that the only valid benchmark for decision support systems is the arithmetic mean of different single benchmark streams, as it is valid for normalized and time-relevant benchmarks [Cro09]. Jain and Raj dedicate several chapters of their book to introduce methods and considerations for metric selection in benchmarking computer systems [Jai91].

3.1. Sensor features

One of the most challenging aspects of benchmarking is selecting the appropriate metrics to be included in the scoring process. In order to identify relevant sensor features for technologies to be applied in smart environments, inspiration is taken from sensor technology overviews [Wil04] and the pervasive model presented by Ranganathan et al. [RAMB*05]. Accordingly, it is possible to identify three different groups of sensor features: sensor performance characteristics, pervasive metrics and environmental influence. These different groups are detailed in the following sections. An overview of the different potential members of this groupsis given, their relevancy for the benchmarking model is discussed and a feature matrix is created that builds the basis for the feature scoring model.

3.1.1. Sensor performance characteristics

This group of sensor features is related to specific technical properties of the given sensing device, as they would be typically put into the datasheet. A first important characteristic is the sensitivity or resolution of a sensor, which is the smallest change of a measured quantity that is still detectable. For example an accelerometer might be able to only detect changes that are above 0.1g. Another important characteristic is the update rate of a sensor. This denotes the number of samples the sensor is able to measure in a certain timeframe. Typically, the number of samples in a second is noted as frequency, thus a sensor may have an update rate of 20 Hz, generating 20 samples in a second. Another factor that is particularly important for embedded systems or wearables is the power consumption of the sensor that may limit the time it can operate on battery, independent of a power source. A last example is the detection range, denoting the maximum distance between the quantity to be measured and the sensing device.

3.1.2. Pervasive metrics

Pervasive metrics can be identified as features that specify how well a given sensor system will perform in collaboration with smart environments, when networked with other devices and when placed into existing surroundings. An example for the latter is the obtrusiveness of a sensor device. If it is clearly visible when applied, if there are disturbing signals generated, or if certain privacy concerns are associated to the sensor device, the acceptance by the user and thus the applicability is reduced. If the sensor is operating in a larger network of other devices, the bandwidth required to submit signal to an analyzing node should be kept low. Equally, if the processing capabilities are limited, less complex data processing is preferable. The overall system cost is increasing if single sensors are particularly expensive, thus limiting the potential applications. The system and attached sensors should be robust, both in terms of physical design and quality of service. Finally, the sensors are more readily applicable if the systems are interoperable to each other.

3.1.3. Environmental characteristics

The third group is the environmental characteristics of a sensor system. Any sensor is affected by a certain disturbance caused by factors in the environment that are similar to the measured quantity, also called noise. For example an optical sensor is influenced by ambient light sources. In this context it is relevant how frequent those influences are in a certain environment and how robust the sensor is against noise. In many cases the presence of noise can be detected and counteracted with a calibration towards the changed environmental factors. The complexity of this calibration is another interesting factor in this regard. Finally, all sensors have some unique limitations, e.g. specific materials that absorb certain wavelengths of the electromagnetic field are difficult to detect for sensors that work in this specific frequency range.

3.1.4. Discussion and feature matrix

To keep the benchmarking constrained the three most relevant features of each category are chosen. This allows a more manageable overall model, however, requires an informed selection of the presented features. Of the sensor performance characteristics group resolution, update rate and detection range are selected. Resolution is a critical feature in any application, determining precise any detection is and if particular objects may be detected at all. Update rate is equally important if fast objects are to be detected and if a reactive systems is desired that respond in real-time. The importance of detection range correlates with the size of the environment and may lead to a reduction of required sensors. Of the mentioned features power consumption is omitted. The actual power consumption of a whole system is a more interesting metric but very difficult to predict from the energy usage of a single sensor [LWG05]. Of the pervasive metrics group unobtrusiveness, processing complexity and robustness are chosen. Unobtrusiveness of the sensor device is a desired feature in many different scenarios, where it should not impede the environment. While microprocessors are becoming ever faster processing complexity is still crucial if the number of sensors is increasing. A dedicated chip will require a more complex architecture and lead to more cost, higher energy usage and more potential points of failure, leading to the final chosen feature of robustness, both against physical abuse, but also in terms of system design, where it is supposed to be resilient towards failure of single components.

Additionally the required bandwidth was omitted, as this metric is not important for many sensors, as they have low bandwidth requirements in general, but also the available bandwidth in wired and wireless systems is increasing continuously. In the last group of environmental characteristics frequency of the disturbing factor, calibration complexity and unique limitations are chosen. If the disturbing factor occurs only rarely it is not critical and therefore should be part of the benchmark. Calibration complexity combines both the processing complexity and time that is required to recalibrate the system. This is highly important in real-time systems that have to monitor the environment continuously. Finally, unique limitations are a rather broad metric that is difficult to quantify. However, in many scenarios it is obvious that a specific limitation might arise, e.g. if the smart environment is in an area with a lot of human noise, microphones could be regularly disturbed. Including this metric allows modeling those applications into the benchmark with a strong weight penalizing unsuited sensors. From the selected metrics a feature matrix is created that allows to associate specific capabilities to a

Table 3.1.: Feature matrix denoting capabilities required for a certain rating

Feature	–	-	o	+	++
Resolution	very coarse	coarse	normal	fine	very fine
Update Rate	less than once per second	slower real-time	real-time	faster real-time	more than 100 times per second
Detection Range	touch	less than one meter	less than 5 meters	less than 20 meters	more than 20 meters
Unobtrusiveness	open large system	open small system	hidden system, large exposure	hidden system, small exposure	invisible
Processing Complexity	single sensor CPU	10+ sensors CPU	single sensor embedded chip	10+ sensors single embedded chip	no further processing
Robustness	single point of failure	error detection	quality of service	self-recovery	fully redundant
Disturbance Frequency	very frequent	frequent	average	unlikely	highly unlikely
Calibration Complexity	very hard	hard	normal	easy	very easy
Unique Limitations	very critical	critical	average	not critical	none

specific rating that will be used later in the scoring process of the benchmark model. Each feature is mapped to five different ratings on an ordinal rating scale comprised of the items “least favorable” (–), “not favorable” (-), “average” (o), “favorable” (+) and “very favorable” (++). This leads to the feature matrix shown in Table 3.1, which will be discussed briefly.

- *Resolution* is ranging from “very coarse” to “very fine”. This unspecific rating is used, as the range may vary strongly between different sensor types. A mapping to actual should depend on the application and object that has to be detected. If the object is large a sensor that would be ranked “coarse” for smaller objects can be ranked as “fine”.
- *Update Rate* is rated around real-time performance that is often rated at around 20 samples per second. Slower sensors might miss various events, while faster sensors allow detecting highly dynamic events.
- *Detection Range* is rated around the 5m distance mark, that is typically enough to cover the entirety of a single apartment room. For larger rooms sensors with a higher detection distance are favorable, many sensors only react to touch.
- *Unobtrusiveness* is ranging from exposed systems placed in the environment (one example would be the Microsoft Kinect) to invisible systems that integrate seamlessly into the environment.
- *Processing Complexity* has a range from dedicated CPUs that are required to process the data of a single sensor to smart sensors that require no further processing, which allows to apply numerous sensors without adding additional processing capabilities to the environment.
- *Robustness* is following criteria for quality of service. The least favorable system fails, if only a single node is present and failing. The preferred system is fully redundant.
- *Disturbance Frequency* is ranging from frequently occurring disturbing signals, to highly unlikely disturbing signals, resulting in a better rating.

- *Calibration Complexity* is a combined metric including the calibration time, the required processing capabilities and if external aid is required in the calibration process, leading to a rating from “very hard” to “very easy”.
- *Unique Limitations* should be ranked according to their criticality, as previously explained they may be suitable to penalize certain sensors or emphasize the prevalence of a disturbing factor in a noisy environment.

Now that the feature matrix is complete, the next step is presenting the formalized benchmarking model and how to use the presented features and their rating to calculate a benchmark score that allows comparing different sensor categories with regard to different applications.

3.2. Benchmarking model

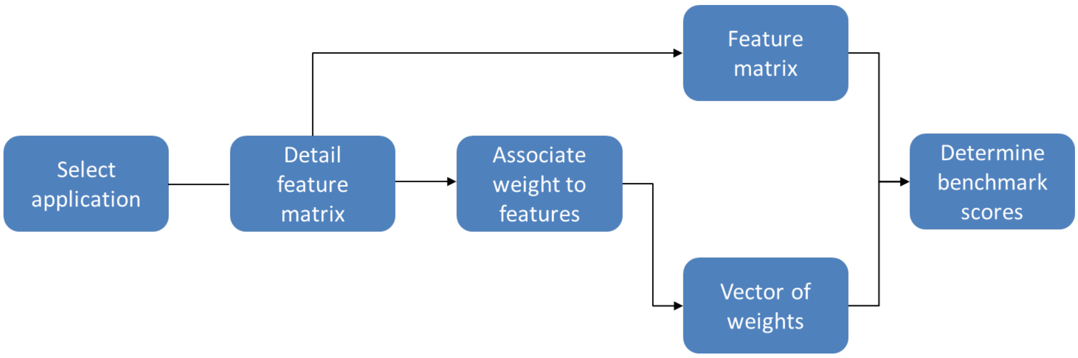

Figure 3.1.: Benchmarking process

In this section I will describe a formal model that will allow us to determine a benchmark score for a given application and a given sensor technology. As previously explained the different applications are distinguished by applying a different set of weights to the known features. As a start the process of this feature weighting is discussed and some examples about proper application are given. Afterwards, the formal model is introduced that deduces a single score benchmark for any sensor technology and any application. The overall process is shown in Figure 3.1 and will be detailed in the following sections, including some example calculations.

3.2.1. Feature score and weighting

The presented feature matrix has some ratings that need detailing in order to be quantifiable in the specific application. The ordinal measurements of the feature matrix should be assigned a quantifiable measure. Taking “Unobtrusiveness” the open system can be detailed as “visible by users” and “large system” as size larger than 100 x 100 x 100 mm. Similar level of details can be applied to the other features leading to the application-specific detailed feature matrix that is used in the scoring process. The different ratings are assigned different numeric values, namely 0.00 (–), 0.25 (-), 0.5 (o), 0.75 (+) and 1.00 (++). The weight of the features for the specific application is also rated on a 5-point ordinal scale, denoted as “not important” (numeric value 0.0), “less important” (0.25), “moderately important” (0.5), “important” (0.75) and “very important” (1.00). Thus for each application there is a distinct detailed feature matrix and a vector of associated weights that can be applied to a set of sensor technologies in order to calculate the benchmark score. The next step is to formally introduce the model that allows to determine a benchmarking score from feature ratings and application weights.

3.2.2. Modeling

The model is supposed to formalize a benchmark for any application and any sensor technology in any domain. As a start the following definitions are given:

- Set of n domains $D = \{d_1, ..., d_i, ..., d_n\}$
- Set of m applications $A = \{a_1, ..., a_j, ..., a_m\}$
- Set of o features $F = \{f_1, ..., f_k, ..., f_o\}$
- Set of p sensor technologies $S = \{s_1, ..., s_l, ..., s_p\}$

In any domain d_i there is have a set of potential applications $A_{d_i} \subseteq A$ and a set of relevant features $F_{d_i} \subseteq F$. For each feature $f_{k,d_i} \subseteq A$ there is the associated feature score $r_{F_{d_i}}$ as explained in the previous section. Each sensor technology s_l has a specific feature score $r_{s_l,F_{d_i}}$ with normalization $\left\| r_{s_l,F_{d_i}} \right\| \varepsilon[0,1]$.

$$\overrightarrow{r_{s_l,F_{d_i}}} = \begin{pmatrix} r_{s_l,f_{1,i}} \\ \vdots \\ r_{s_l,f_{o,i}} \end{pmatrix} \tag{3.1}$$

The weights w_{f_o} associated to a specific application a_j in a domain d_i have the same cardinality $|w|$ as the vector of feature scores $\left|\overrightarrow{r_{s_l,F_{d_i}}}\right|$. The values are determined, so $\|w_{f_o}\| \varepsilon[0,1]$.

$$\overrightarrow{w_{a_j}} = \begin{pmatrix} w_{f_1,a_j} \\ \vdots \\ w_{f_o,a_j} \end{pmatrix} \tag{3.2}$$

The feature scores and associated weights allows to determine a benchmark score b_{s_l} for a specific sensor technology s_l for any application a_j by using the scalar product of feature score and respective weight and apply normalization with regard to the weight.

$$b_{s_l} = \frac{\overrightarrow{r_{s_l,F_{d_i}}} \cdot \overrightarrow{w_{a_j}}}{\sum_{k=1}^{o} w_{f_k,a_j}} \tag{3.3}$$

Thus it is possible to compare different sensor technologies by calculating and comparing the different benchmark scores for a given set of sensor technologies $S_p \subseteq S$ and receive a set B_{S_p} with $t = |s_p|$.

$$B_{S_p} = \{b_{s_l,1}, \ldots, b_{s_l,t}\} \tag{3.4}$$

In order to determine the optimal (chosen) sensor technology b_c for an application a_j and given the prerequisites regarding non-negativity of weights and feature scores we can evaluate the set for the maximum element.

$$b_c = max(B_{S_p}) \tag{3.5}$$

3.2.3. Feature score normalization

With regards to actual benchmarking the problem of bias towards a specific technology may occur. If the average features ratings are different between two technologies the calculated benchmark score will increase. In many instances this might be beneficial, yet if comparing numerous technologies to a set of different applications a trend might be more important than absolute scores. Thus, we provide an optional step of calculating the normalized feature vector $r_{s_l,F_{d_i},norm}$ with regard to the average associated value of 0.5, using the following equation.

$$\overrightarrow{r_{s_l,F_{d_i},norm}} = \begin{pmatrix} r_{s_l,f_{1,i}} \\ \vdots \\ r_{s_l,f_{o,i}} \end{pmatrix} \cdot \frac{o \cdot 0.5}{\sum_{p=1}^{o} r_{s_l,f_{p,d_i}}} \tag{3.6}$$

The feature-normalized benchmark score is accordingly determined with the following equation.

$$b_{s_l} = \frac{\overrightarrow{r_{s_l,F_{d_i},norm}} \cdot \overrightarrow{w_{a_j}}}{\sum_{k=1}^{o} w_{f_k,a_j}} \tag{3.7}$$

3.2.4. Scoring

Now with the formal model and the available set of feature matrix and weights we are able to calculate the benchmarking score for a set of sensor technologies. As an example we are choosing the application indoor localization in a public shopping area to monitor customer behavior. The requirements include a tracking accuracy of 50 cm, with a large area to cover and potentially fast moving persons. Thus the importance ratings for performance characteristics are moderately important for resolution, important for update rate and very important for detection range. The system can also be used for security purposes, thus unobtrusiveness is less important. There can be dedicated servers, so processing complexity is not important, but the system should be difficult to disturb, thus robustness is important. Disturbance frequency is moderately important, as a large number of per-sons is monitored, leading to statistically significant results, even if single measurements are disturbed. The environment is fairly static, thus calibration complexity is less important. It is possible that a crowded shop produces a lot of acoustic noise, therefore no unique limitations towards acoustic disturbances should be present and this is moderately important. The resulting vector of weights is:

$$\overrightarrow{w_a} = \begin{pmatrix} 0.50 & 0.75 & 1.00 & 0.25 & 0.00 & 0.75 & 0.50 & 0.25 & 0.50 \end{pmatrix}^{-1} \tag{3.8}$$

Based on previous experiences and best practice a camera-based system is evaluated. The system has high resolution cameras, with an update rate of 30 samples per second and a high detection range of more than 20 meters. The cameras are external, not hidden from view but attached on the ceiling. The processing complexity is very high, requiring a dedicated system per camera. Since they are out of reach they are robust towards human intervention and independent from each other. In the given setting visual disturbance is unlikely, calibration is difficult but not required regularly and the system is not disturbed by acoustic noise. This results in the following rating vector:

$$\overrightarrow{r_{s,f}} = \begin{pmatrix} 1.00 & 0.75 & 1.00 & 0.25 & 0.00 & 0.50 & 0.75 & 0.25 & 1.00 \end{pmatrix}^{-1} \tag{3.9}$$

Using those two vectors we can calculate the final scoring for this sensor system using the equations of the previous section, leading to $b_s \approx 0.78$ and a feature-normalized score of $b_{s,norm} \approx 0.63$. Determining the feature rating vector for other technologies is possible in a similar fashion. The optimal technology would have the highest score b_s or $b_{s,norm}$.

3.3. Evaluating technology popularity

In order to evaluate the method we propose a discussion based on previous successful works in the domain of smart environments. We will select three different application areas and for each benchmark three different sensor technologies. In order to estimate how popular a certain technology is for a given application we will be using the ACM Digital Library to query scientific publications with respective author keywords. This method is limited, as the chosen keywords may not catch all relevant publications. Therefor we will slightly increase the focus by using multiple associated search terms for each application and technology. Additionally we will also perform respective searches using the Google Scholar data-base that has a much broader scope-The advantages of the latter are the huge collection of scientific resources and no strong selection bias. However, there are various associated issues that may affect the method. The search results vary on the search term, additionally there will be results that mention the search term but do not necessarily rely on the technology for their respective system. Therefore, the results should be considered as an indicator for popularity in the research community. Similar to the ACM DL search we are also looking for synonyms and calculate an average between the search results.

As applications we choose hand gesture interaction, a marker-based identification system and obstacle avoidance for an autonomous system. The technologies are camera systems, radio-based systems, depth or stereo cameras and ultrasound devices.

3.3.1. Scoring

Table 3.2.: The importance weighting of different applications, based on the features.

	res	upd	det	unob	proc	robu	disfr	calco	uniqd
Hand Gesture	++	++	-	-	-	o	+	o	o
Identification	–	-	++	++	o	++	+	-	+
Obstacle Avoidance	-	+	-	o	+	+	++	++	+

At first we determine the weights of the different applications with regards to the features. The results are shown in Table 3.2. For the tables in this section we are using short notation of the features in order of appearance in Section 3.2.1. The rating of the different technologies and the resulting score is shown in Table 3.3. Here

Table 3.3.: Feature rating of the different sensor technologies

	res	upd	det	unob	proc	robu	disfr	calco	uniqd
Camera	++	o	+	-	o	o	o	-	o
Radio	-	+	++	+	o	o	o	o	-
Depth camera	+	o	o	-	-	o	-	o	o
Ultrasound	-	+	o	o	+	o	+	o	o

it is possible to follow different strategies regarding the rating. In terms of unbiased comparison looking at the equations it would be necessary that all technologies have the same average feature rating. The second strategy is to apply an absolute ranking to all technologies, independent of the given application. This might lead to certain technologies being unsuited for a given task, or technologies that have the best benchmark score regardless of application. In this specific case the average rating is 0.53 for cameras, 0.58 for radio, 0.44 for depth cameras and 0.56 for ultrasound devices. Table 3.4 displays the different calculated benchmark scores for the combinations between applications and technologies. As we are comparing numerous technologies and applications the feature-normalized benchmark score is also included. The effect of the normalization is easily visible. Particularly radio has a high feature rating and is negatively affected by the normalization. The only

Table 3.4.: Regular and normalized benchmark score matrix of different applications and technologies

		Camera	Radio	Depth Camera	Ultrasound
Hand Gesture	b_{s_l}	0.58	0.53	0.49	0.55
	$b_{s_l,norm}$	0.54	0.45	0.55	0.50
Identification	b_{s_l}	0.49	0.64	0.40	0.57
	$b_{s_l,norm}$	0.46	0.55	0.45	0.51
Obstacle Avoidance	b_{s_l}	0.47	0.56	0.42	0.59
	$b_{s_l,norm}$	0.44	0.48	0.47	0.53

example with a negative average feature rating is the depth camera. After applying the normalization it becomes competitive in some applications.

Finally, Table 3.5 shows the search results regarding the different technologies and applications. Particularly the ACM DL keyword search can generate empty results if the search terms are too specific. Thus, the search terms we were using are “gesture“, “identification and “obstacle” in this regard and add synonyms for the different technologies. For each sensor category we allowed the following synonyms. “Camera” and “video” for the first technology, “radio”, “rf” and “wifi” for the second, “depth camera”, “stereo camera” and “Kinect” for the third and “ultrasound” as well as “ultrasonic” for the last one. All search results were averaged according to the number of synonyms used For the Google Scholar search we used more specific terms, “hand gesture”, “user identification” and “obstacle avoidance” with the same synonyms to prevent an excessive number of search results and prevented inclusion of patents and citations. All searches were performed on January 30th, 2014.

Table 3.5.: Search result frequency given specific applications, sensor technologies and synonyms for ACM Digital Library (DL) and Google Scholar (GS)

	Camera		Radio		Depth Camera		Ultrasound	
	DL	GS	DL	GS	DL	GS	DL	GS
Hand Gesture	66	14100	27	7350	32	6850	3	1660
Identification	81	5590	162	4920	10	3957	5	599
Obstacle Avoidance	8	24000	1	13017	17	12278	8	14500

3.4. Discussion

In this evaluation we included both benchmark score types to outline their dif-ferences. “Camera”, “radio” and “ultrasound” have a feature rating above average, whereas “depth cameras” had a lower than average rating. The feature-normalized benchmark score is thus adapted accordingly. Regarding the application of “hand-gesture recognition” this leads to “depth cameras” being considered the optimal technology as opposed to “cameras” that had a higher score before normalization. For the other applications there is no change in optimal technology. The preferred strategy for applying feature-normalized or non-feature-normalized benchmark scoring should depend on the specific benchmarking process. If we are bench-marking numerous technologies and applications in a single process the feature-normalization might be helpful to get a tendency regarding the optimal system.

However, if the application is very specific it might be preferred to get a clear ranking and penalize unsuited technologies, regardless of their average feature weight. Accordingly, it is possible to refrain from normalization. Looking at the search results we can draw several conclusions. The prevalence is unequally distributed between the different technologies. Both in keywords and general occurrence cameras are the most commonly occurring sensor device, with radio and depth camera ranked behind. Ultrasound on the other hand is less frequently occurring. This may be explained by the higher versatility of the other options. Regarding the “hand gesture” application, cameras have both the highest benchmark scores and most results in the database searches. The benchmark score for “user identification” and “radio” are matched for the ACM DL. However, there are more GS results for “camera”. As already mentioned cameras are more com-monly used, yet, the difference in keyword search results is significant. “Obstacle avoidance” is least common in the ACM DL, however quite popular in GS. Accordingly, “ultrasound” sensors are significantly more common in both searches, as opposed to the previous applications. Nonetheless, “stereo cameras” are the most common sensor device for this application. They are commonly used in automotive scenarios, where the detection range of ultrasound is insufficient, as the objects are moving fast [BB98]. Therefore, the application scenario might have to be redefined for fast-moving object detection in open areas as opposed to obstacle avoidance for robots in home scenarios. We additionally have to discuss the method of using database searches for verifying the benchmarking method, as opposed to expert opinion. Surveys of a specific application or certain technologies are common in scientific literature. How-ever, while they might be comprehensive and cite several hundred different applications, the ACM DL database covers more than 2.2 million entries and GS searches can lead to more than 9.7 million results. Therefore, the index searches are preferable in terms of broadness. The search for keywords in ACM DL results in few hits compared to the database size. As they are chosen by the authors there is a large variety in word choice, spelling or number of keywords. While extend-ing the number of different searches might lead to more results overall, it may also lead to additional overshoot, including work that do not cover the desired topics. The GS searches are very prone to overshooting, and should be preferably used to discover trends in data, as opposed to narrowly clustered results. Finally we want to briefly discuss the tendency of the benchmark scores to crowd around 0.5. While the benchmark may result in any number between 0 and 1 the two normalization processes and the tendency to avoid extreme scoring for features and weights leads to an average close to 0.5. Thus, even slower differences close to this average may have a higher significance. It would also be possible to put more emphasis by using a correction term.

3.5. Applications for capacitive proximity sensors

4. Application prototypes

4.1. CapFloor

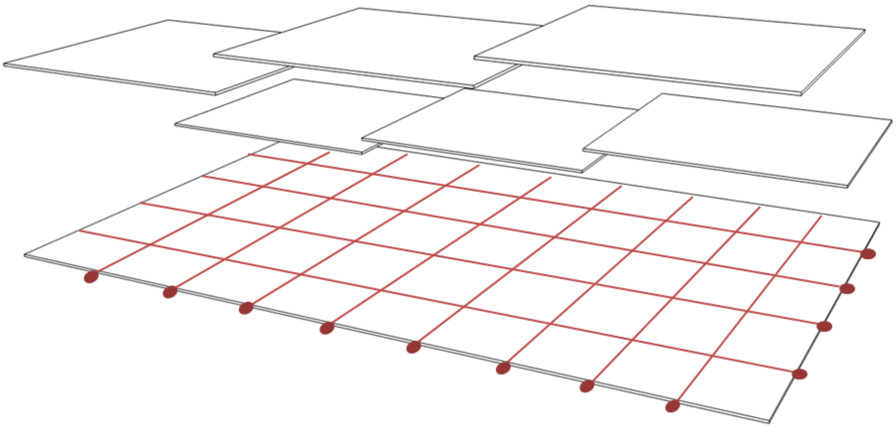

Figure 4.1.: CapFloor sketch - grid layout of electrodes is placed below a floor layer with sensors attached on the sides

CapFloor is a capacitive system for indoor localization and fall detection that is based on a grid array of sensing electrodes placed below a floor covering [BHW12]. A sketch of the system is shown in Figure 4.1. The grid is comprised of insulated wires that are placed orthogonal to each other. Sensors are placed on two sides of the room. Each sensor is performing loading mode measurements. The system is intended to act as both indoor localization system and fall detector. CapFloor can be placed below any non-conductive material, like wood, tiles and PVC, if the distance between the wires and the floor surface is not too high. It can discriminate between a foot being above an electrode or a whole body. Combining this information from various sensors we are able to get a reliable detection of lying, sitting and standing persons. Using only two sides of the room for sensors it is possible to cut the wires without considerably affecting the signal; allowing easy installation in non-rectangular rooms. Accordingly CapFloor is able to be used in various application scenarios. Indoor Localization in the home domain can be useful in energy saving and fall prevention by appropriately activating and deactivating the environment lighting. It can also be used in security-restricted areas to detect unauthorized movement. The fall detection should be used in a system that has various levels of escalation. E.g. it is not easy to distinguish between a person doing exercises on a floor and a person that has fallen down. Accordingly the system should query if the person is well and not autonomously call for outside help.

4.1.1. Data processing

Using long wire electrodes may result in considerable noise and influence from outside electric fields. Therefore CapFloor requires preprocessing to reduce the noise and achieve a more robust high-level data processing. The localization uses the weighted average algorithm that has been presented previously. The fall detection is using

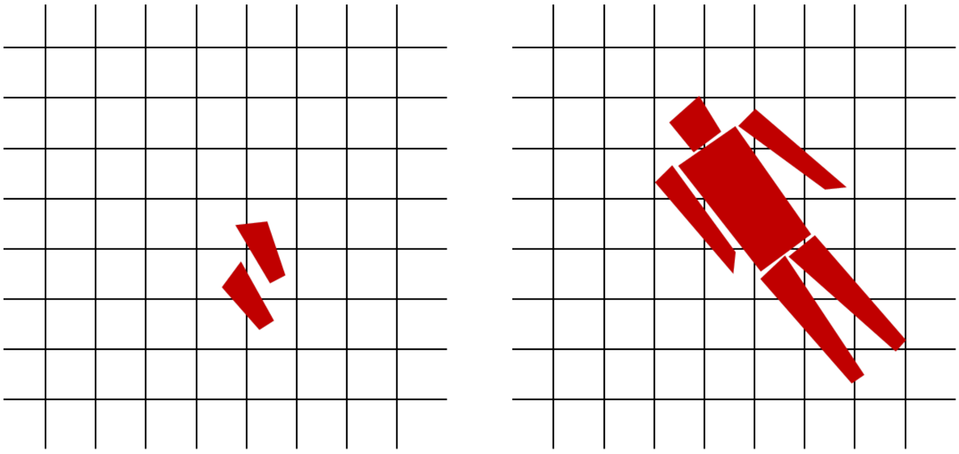

Figure 4.2.: Shapes of a standing and lynig person on top of the CapFloor grid

a time-series analysis of the aggregated values of the sensors that are currently detecting an object. This method is using the assumption that the overall sensor response is roughly equivalent to the shape of the object that is closest to the surface, resulting in a higher capacitance of the overall system, similar to the plate capacitor model. This effect is shown in Figure 4.2. The sum s of all n sensor values r is the closest equivalent to the system capacitance and therefore a viable measure. If the overall value is beyond a certain threshold v_l we can consider a lying person p_l.

$$s = \sum_{i=0}^{n} r_i \quad , \quad p_l = \begin{cases} 1, & s \geq v_l \\ 0, & s < v_l \end{cases} \tag{4.1}$$

In order to increase the robustness this threshold has to be exceeded for a certain amount of time t_m. In consequence a fall f is detected if the following equation is 1.

$$f = \prod_{j=0}^{t_m} p_{l,t_j} \tag{4.2}$$

4.1.2. Evaluation

The CapFloor system was evaluated in the scope of the Indoor Localization Track of EvAAL 2011, where it participated out of competition [CK12]. In Figure 4.3 we can see a picture of the demonstration setup installed in the living lap using the system integrated into different mats that are placed in the environment. The system was tuned to detect a single person and was able to perform this reasonably in the areas covered. The resolution of the system is strongly depending on the given density of electrode wires. While there is a certain measure of proximity, it is not possible to detect objects that are more than a few centimeters away from the wires. Later iterations of the system are using higher voltages and shunt mode measurements to improve the tracking reliability and enhance the fall detection.

Figure 4.3.: Floor mats with integrated CapFloor system used at the EvAAL 2011 competition [BHW12]

4.2. Smart Bed

The Smart Bed is a regular bed frame that has been equipped with capacitive proximity sensors in order to determine occupation, posture and sleep phases [BH12] [DBM13]. A sketch can be seen in Figure 4.4. The electrodes are comprised of copper foil that is attached to the flexible wooden panels of the slatted frame. This allows the electrodes to be sensitive to both proximity and applied pressure, resulting in a superposed combined sensor value that is considerably higher as opposed to proximity measure on its own. The electrodes are equally distributed, with four being on both sides of the two person bed. The system is able to determine different sitting and lying postures of one or two persons, including less regular lying positions such as diagonal or orthogonal to the long side of the bed. Using an analysis of the movement gathered by variation in the sensor signal the sleep phases can be analyzed, similar to accelerometer-based systems that are popular for smartphones [KJJ11].

The Smart Bed can be used for various purposes. A main application is connecting the occupation detection to a home automation system and timer in order to activate ambient lighting if the person is get-ting up in the night, presumably to find the way to the restroom. Accordingly, in a single person household the lights in unoccupied rooms could be turned off in order to conserve energy. In the domain of personal health the Smart Bed is able to give the user a feedback on sleep quality based on the sleep phase measurement performed in the night. Another potential application is to use the acquired pressure distribution as indicator for back-friendly lying positions that may be harmful over a longer period of time [HB10]. The occupation and posture detection relies on a simplified body model to approximate the pressure distribution and sensor values to a certain posture [BH12].

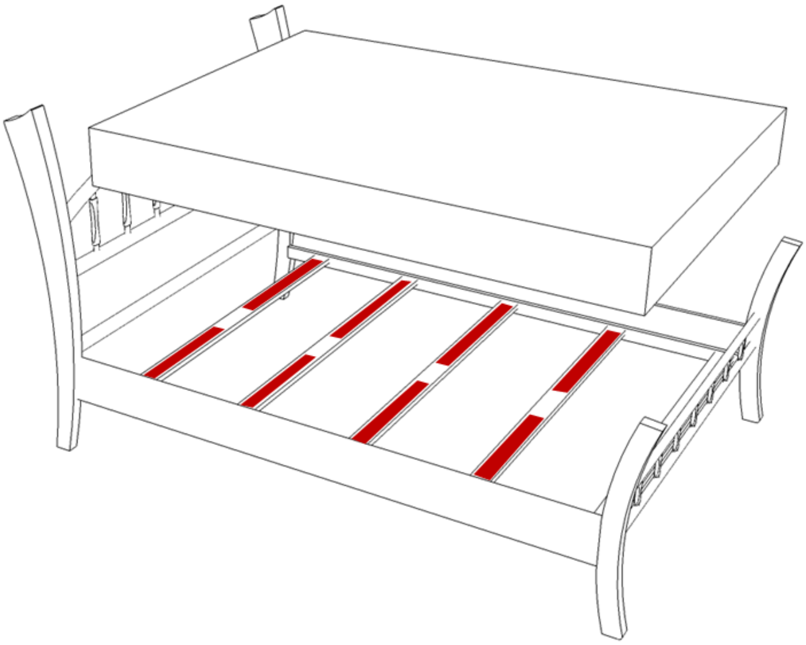

Figure 4.4.: Smart Bed sketch - flexible plate electrode are attached on spring board

4.2.1. Data processing

The different components of the Smart Bed data processing are shown in Figure 4.5. Raw sensor data is distributed to three different modules, the calibration which is determining the initial parameters for the sensor data fusion, the drift compensation that alters those parameters according to long term trends and finally the sensor data fusion module that processes the data and does feed it to the occupation & position detection. Calibration and drift compensation follow the previously presented model [BH12]. Occupation and position detection is performed by dividing the two person bed into left and right and individually calculating for each side the total sensor values, assumed center of pressure using weighted average and the standard deviation (Figure 4.6). The same calculation is done between the two sides to distinguish where is activity or if one person is lying diagonally. Using these six intermediate values we can now map various poses. If all activity is on one side and the horizontal deviation is low, we can assume that one person is sitting. We can additionally use the intermediate values to calculate more information, e.g. the exact location a person is sitting at. The data processing for the sleep phase recognition is based on detecting the sensor data variations in order to analyze movement. Discriminating between sleep phases using movement is a common approach that has been used in the past [SL86]. Using a sparse set of sensors it is possible to detect movement by comparing subsequent sensor readings and associate it to different sleep phases using different activity profiles. The system is based on the same prototype as the posture recognition system [DBM13].

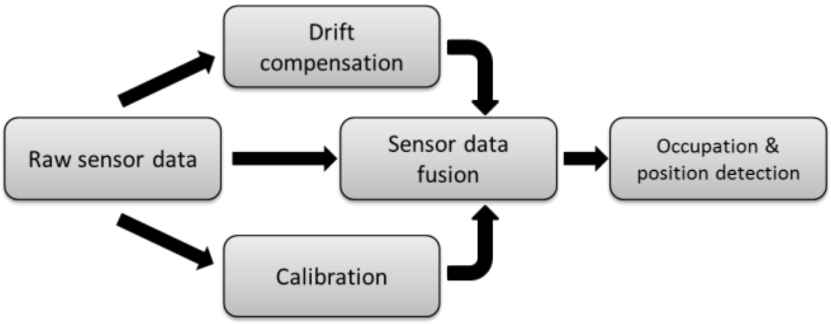

Figure 4.5.: Data processing components [BH12]

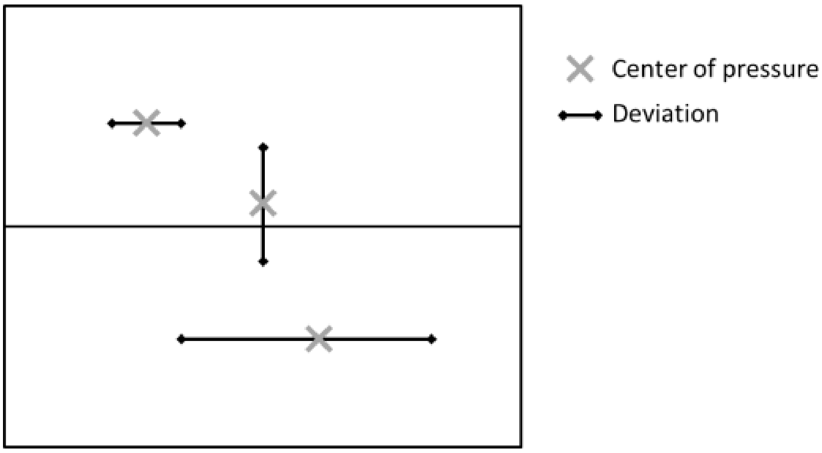

Figure 4.6.: Calculating centers of pressures and deviation [BH12]

4.2.2. Evaluation

The Smart Bed posture recognition is able to successfully distinguish eight typical sitting and lying states. Using adaptation of the intermediate values it is possible to fit the state to an actual position on the bed, e.g. a *person sitting on the right side of the bed* state can be modified to any location on that specific side of the bed. Regarding the detection of sleep phases there has been an evaluation and benchmarking of three nights [DBM13]. The Smart Bed was able to achieve a comparable performance to smartphone applications that detect sleep phases based on accelerometers. Figure 4.7 gives an example of movement recordings using the capacitive proximity sensors over one night. The activities are grouped into distinct chunks that are later associated to the sleep phases. Currently breathing rate detection is added to the Smart Bed that can be used to improve the sleep phase detection and also can potentially detect anomalies that may be indicative of a certain health risk.

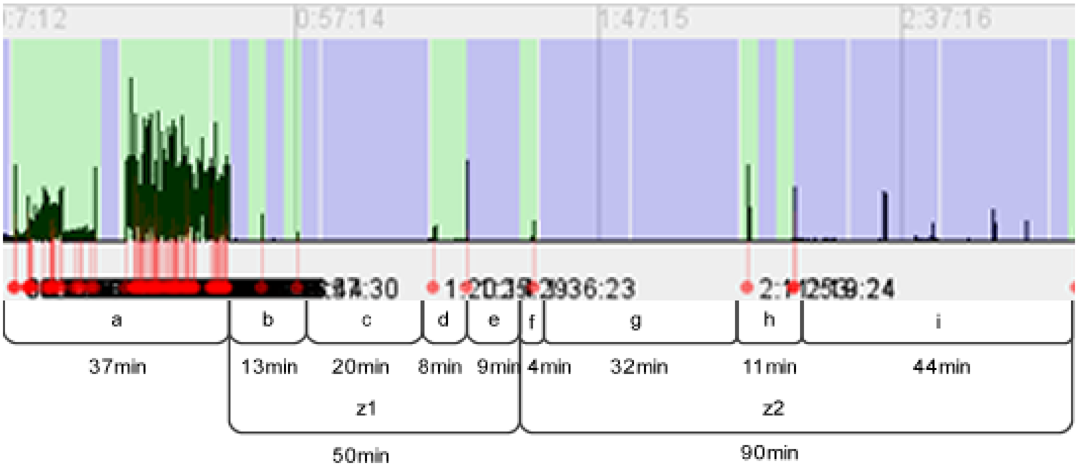

Figure 4.7.: Sleep movement data over three hours in one night [DBM13]

4.3. The Capacitive Chair

The Capacitive Chair is a regular office chair equipped with eight capacitive proximity sensors that can detect different sitting postures and work-related stress levels by examining movement and breathing rate [BF13]. Seven solid copper electrodes that are placed below the covering are augmented by a single conductive thread electrode that is placed in a mesh on the backrest. In the past smart chairs have used pressure sensors to infer posture and occupation [TSPM01]. Combining presence and proximity sensing it is possible to directly infer postures where parts of the body do not touch the surface, e.g. if the body is arched towards the front, or if an arm is raised from the armrests. Additionally higher area electrodes in the backrest allow detecting the breathing rate by measuring the movement of the chest.

The Capacitive Chair aims at providing different services to a typical office worker and office managers. Using the occupation detection it is possible to advise for some type of physical activity, if the time spent in front of the screen was too long. The system can also advise the user to change to a more back-friendly posture or regularly switch the stance to achieve a more general workout. Using the breathing rate detection we are able to get some sort of measure of the current stress level associated to the given working situation. By adapting the environment it is possible to improve the working atmosphere and reduce stress. The Capacitive Chair uses a multifacetted data processing approach. A machine learning algorithm is associating the sensing data to one of nine different typical sitting positions, inspired by a recent study of sitting positions for modern device usage [Inc13]. An adaptive body model that is fitted to the current sensor values allows for fine grained adaptation of those postures. Finally a combination of Fourier and data variation analysis is calculating the current breathing rate [BF13].

4.3.1. Data processing

In Figure 4.9 we can see a screenshot of the Capacitive Chair debug application. On the left side we see a 3D model that is fitted to a chair model according to the current sensor values, in the middle the results of the machine learning module and the recognized posture and on the right side the currently running breathing rate detection as both Fourier analysis and signal deviation analysis. All processing methods work on filtered and normalized sensor data. The difference in shape, material and size of the electrodes necessitates slight adaptations

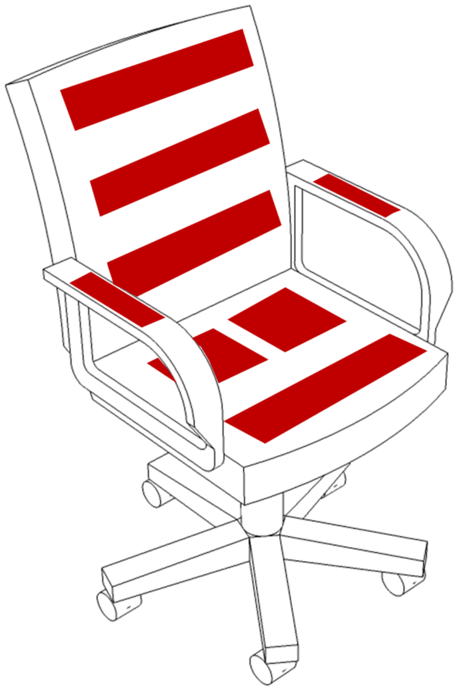

Figure 4.8.: Smart office chair sketch - eight electrodes three in backrest, three on seat and two in armrests

to noise filtering and data processing. As an example only the conductive thread backrest electrode is used in the breathing rate detection. The 3D model is using a simplified human joint model comprised of 13 connected components. Based on the current sensor readings, single parts or groups of components are fitted to the virtual chair. The process is a mix of posture mapping as found in the smart bed and modification of the dynamic links between the single components [BF13]. We use a simple RBF neural network and training data collected by two different persons to match the input from eight sensors to nine potential output postures that are associated to different working situations. An early observation is that certain postures are difficult to distinguish given the limited number of sensors and the similarity of the postures on the rigid chair. Either a higher number of sensors or a more versatile chair could be used that allows gathering additional information required to distinguish the different poses more reliably.

The breathing rate detection is operating on a single electrode that is integrated into a mesh on the backrest using conductive thread. The setup is shown in Figure 4.10. Consequently the surface of the electrode is large and able to pick up the chest movement. Two different methods of data processing are used and fused to get the final breathing rate. Using a fast Fourier transformation the signal is transformed into the frequency space. We are looking for significant signal portions in frequency areas that can be associated to breathing, between $0.2Hz$ and $10Hz$. The second method is to look for zero-crossings of the sensor signal through an adaptive baseline. If a person is breathing in the sensor value will decrease resulting in the signal dropping below the long-term average, and rise above when the person is breathing out. Accordingly the breathing rate can be calculated by counting the zero-crossings.

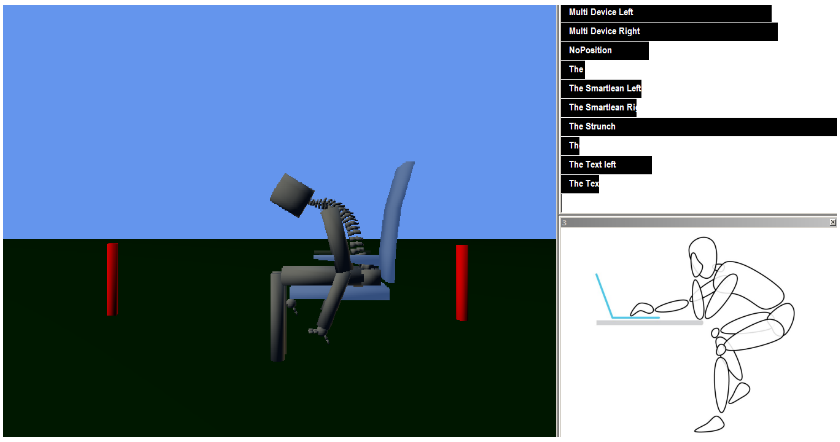

Figure 4.9.: Screenshot of the Capacitive Chair application showing the fitted 3D model on the left, posture detection on the upper right and the recognized posture on the lower right

4.3.2. Evaluation

4.4. Active Armrest

The Active Armrest is a prototype to demonstrate unobtrusive gestural interaction in the domain of automotive applications [74]. The interior of modern cars can be considered a smart environment as it includes an ensemble of sensors and actors that adapt the system behavior according to user preference.

Many cars use touch screens or jog dials to control primary and secondary car functions [SDKS10]. Capacitive proximity sensors allow integrating interactive areas into different existing surfaces of a car, e.g. an armrest. The Active Armrest is using a set of eight sensors that are separated into two different groups. There are two larger electrodes in the back of the armrest that are dedicated to recognizing the presence of an arm. In the front of the armrest there is an array of six small electrode sensors, in order to register finger gestures, as shown in Figure 4.11. The basic idea is to disallow interaction when the arm is resting and enable it once it's lifted. The Active Armrest supports swiping gestures of a single finger and static holding gestures of two fingers. This allows controlling various typical automotive applications, e.g. a navigation application, whereas holding is zooming in and out and swipe pan through the maps. Similar applications, such as multimedia features and comfort settings can be controlled in a similar fashion.

4.4.1. Data processing

As we already mentioned, the Active Armrest electrodes are put into two groups. The data processing for both groups is distinctly different. In order to detect the presence of the arm using the two-electrode group a simple threshold on the accumulated values is used. The six sensor array in the front (touch area) is using the presented weighted average method to calculate finger positions. Additionally a threshold is used to distinguish one and two fingers. Overall there is a data processing pipeline as shown in Figure 4.13. The finger tracking and gesture recognition will be inactive until it is ensured that no arm is present.

Figure 4.10.: Screenshot of the Capacitive Chair application showing the fitted 3D model on the left, posture detection on the upper right and the recognized posture on the lower right

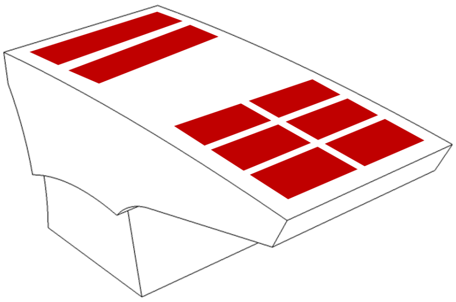

Figure 4.11.: Active armrest sketch - six electrodes for finger gesture detection in front, two for arm detection in back

4.4.2. Evaluation

In order to evaluate the Active Armrest we have built the prototype shown in Figure 4.12. An aftermarket armrest was equipped with an OpenCapSense toolkit. The demonstration application is based on the SenseKit debug software supplied with the toolkit. As of now there is a simple USB connection to a nearby PC. Figure 4.14 shows a screenshot of the finger tracking application on the left, with a two-finger touch registered on the upper left part of the touch area. It is interfaced with a TUIO [KBBC05] based maps application using OpenStreetMap [HW08] data. The map is moved around using simple swipe movements of the finger that are directly associated to pan-features of the demonstration application. Zooming is activated by two-finger hold gestures on the upper or lower part of the touch area. We have used public displays of this prototype to get an idea of how easily unaffiliated persons learn to use the system. While the majority agreed on the potential of the application, there have been some reservations regarding the current gesture set, particularly that a closer relationship to smartphone touch screen gestures would be welcome.

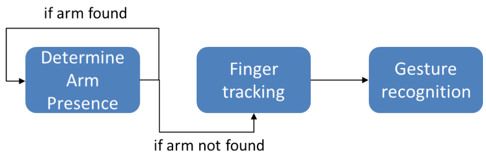

Figure 4.12.: Data processing pipeline of Active Armrest

Figure 4.13.: Active Armrest prototype, left - outside view, right - detail view of electronics

4.5. Magic Box

The so-called MagicBox was our first attempt to create an interaction device based on capacitive proximity sensing. It is using an array of six individual wireless capacitive sensors that communicate to a central station [BH11]. The electrodes are using a large surface area and are made of aluminum foil. A sketch is shown in Figure 4.15. The system is able to track the position of a single hand in three dimensions up to a distance of approximately 20*cm*, and uses different methods to infer gestures from the hand movement. It is designed to be a generic interaction device that can potentially be hidden below non-conductive surfaces. As it can be used without touching it is also applicable in sterile environments. A suite of demonstration applications has been created that showcase typical scenarios for the MagicBox. This includes multimedia applications, like image viewer and media player but also a 3D object viewer intended as demonstrator for potential medical applications, allowing a surgeon to check MRT or CT images in a sterile environment without touching any surface.

4.5.1. Data processing

The first data processing step of the MagicBox is the planar localization of the hand, following the weighted average algorithm previously presented. In order to calculate the distance of the hand from the plane we are using a piecewise linear interpolation, that resembles the response curve of a single sensor [BH11]. An addition of the MagicBox was a generic gesture recognition module based on methods similar to mouse gesture recognition [BDK13], albeit adapted for three dimensional locations. The developed debug software allows defining an

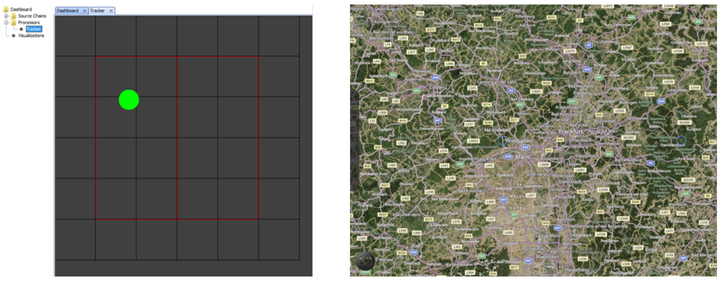

Figure 4.14.: Active Armrest demo software, left - finger tracker, right - OSM based navigation application

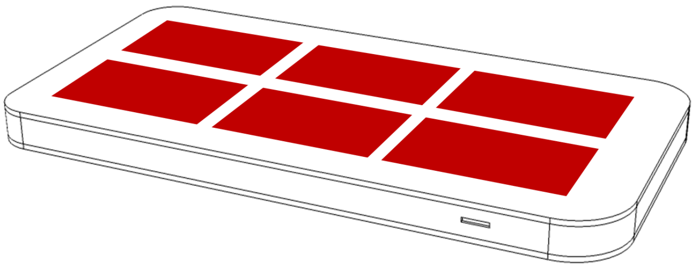

Figure 4.15.: MagicBox sketch - six electrodes uniformly distributed below surface

arbitrary set of potential gestures and adding training data, as shown in Figure 4.17. The module is looking for matches based on the most recent set of locations.

4.5.2. Evaluation

The MagicBox prototype is based on the Cypress First Touch starter kit [Cor13] and combines six capaci-tive sensors communicating wirelessly to a single base station, that are put together with a USB-rechargeable power supply into a casing. A concep-tual rendering showing the interaction area and a detail view of the prototype electronics are shown in Figure 4.18. The different iterations of the MagicBox have been evaluated in conjunction with various demon-stration applications. A usability study with 18 per-sons led to general approval of the system [BH11]. Two of the applications used in this study are shown in Figure 4.19. On the left is a 3D object viewer that has to be controlled by a combination of menu and direct manipulation of the screen content. On the right side there is an image viewer that was controlled by gesture to trigger the next/previous images or perform zooming operations. The most common positive remarks gathered in this study can be roughly put into three groups:

- The device very intuitive to use
- The idea of interacting this way is novel and interesting
- It is easy to control applications with those gestures

Likewise we identified three main groups for negative comments about the prototype:

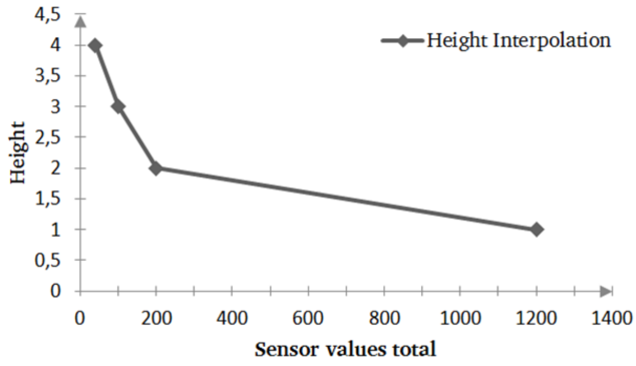

Figure 4.16.: Piecewise linear hand distance estimation [BH11]

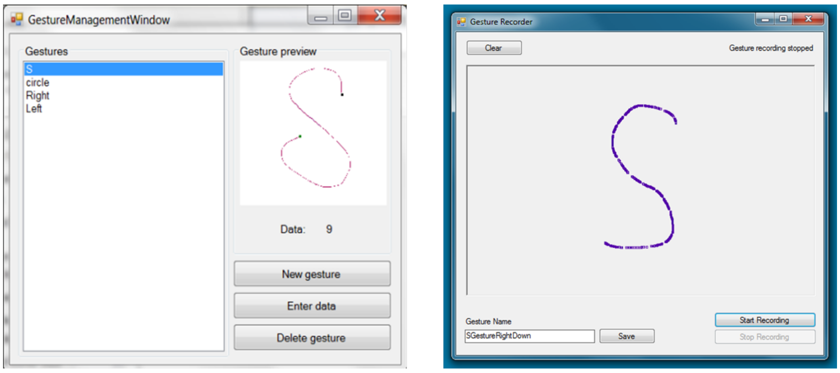

Figure 4.17.: Gesture overview module (left) and gesture recorder (right)

- The device is not very precise
- The interaction speed is slow
- It can be tiring for the arm

Later iterations have been trying to improve some of the weaknesses presented above, e.g. by using a more sophisticated gesture recognition system and faster sensor refresh rates. Accordingly there were fewer complaints about interaction speed and precision [BDK13]. However, the final complaint about the device being tiring for the arm, requires a different approach, that we are investigating in the final prototype to be presented in this system.

4.6. CapTap

The CapTap is a large area interaction device unobtrusively integrated into a living room table. It is comprised of 24 capacitive sensors and a single sen-sor for knock detection that supports selection events within the demonstration applications [80]. In the domain of free-air gestural interaction there are two prevalent challenges. The

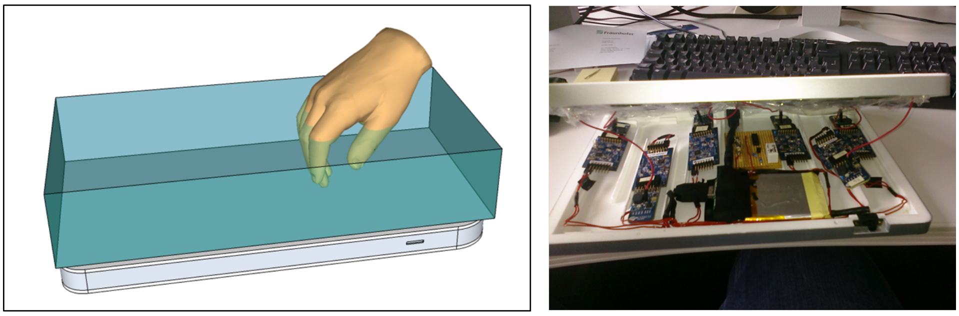

Figure 4.18.: MagicBox conceptual rendering (left) and detail view of electronics (right) [BH11]

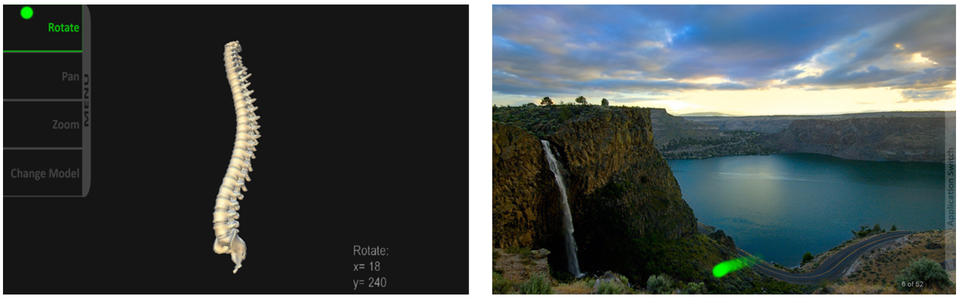

Figure 4.19.: MagicBox demonstration application - 3D object viewer (left) and image viewer (right) [BH11]

physical demands of pro-longed interaction with such systems is high [81], [82]. Additionally it is difficult to adapt selection events to gestural input. The latter is typically real-ized using time- or position-based gestures [81], [83]. There is no trivial solution to these challenges and any approach has to take into account the specific application scenario covered. Several systems are trying to provide specific GUIs, while others include additional input devices assisting the interaction [84], [85]. CapTap presents an approach to improve the interaction speed of invisible input devices based on capacitive proximity sensors. We have developed a method to unobtrusively detect knocks on a table equipped with a hand tracking system based on capacitive proximity sensors that allows emulating selection events that would typically require an additional time- or movement-based gesture.

4.6.1. Data processing

The hand location of the CapTap is similar to the methods presented for the MagicBox. We add the additional component of knock detection to provide selection events when touching the surface. Figure 4.20 shows a sketch of the knock detection system. The table has a glass plate that is suspended on some rubber supports. In the center of the table we attach a small peg (enlarged in sketch) that creates a connection between the glass plate and a piezo sensor. If the glass plate starts vibrating from a touch we can measure this using the piezo sensor [BF13]. If a notable vibration is measured we are collecting the next 50 samples, resulting in a window of 250 milliseconds. To distinguish single and double knocks we calculate the weighted average within this window to get a measure for the distribution of sensor values within. If the average is closer to the beginning of the window the resulting event should be a single knock, and a double if the average is closer to the end of the window. Hand localization and knock detection are working independently and are combined later in the

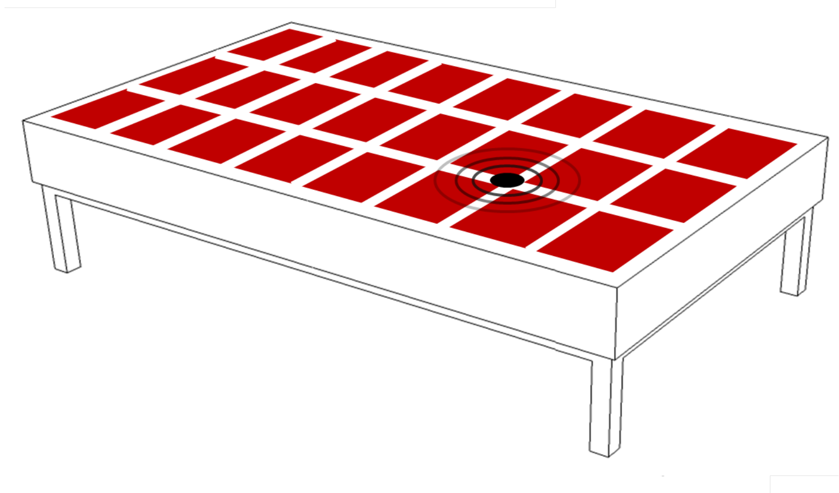

Figure 4.20.: CapTap sketch - 24 electrodes placed under table surface

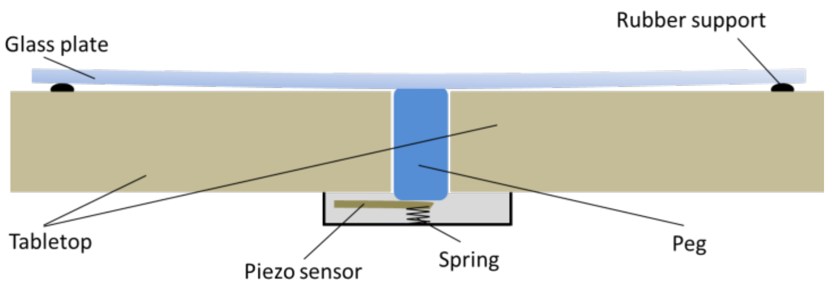

Figure 4.21.: Suspended peg knock detection system for CapTap [BF13]

software. It is reasonable to combine this, e.g. to ignore knock events that are occurring without a hand present. They may be indicative of a person doing a strong step close to the table.

4.6.2. Evaluation

The CapTap prototype is integrated into a common living room table. Some photos can be seen in Figure 4.22. On the left side we see the 24 electrodes made of non-etched circuit boards. A sensor is attached to each. The knock detection box with fixation, housing and piezo sensor is shown on the right side. The overall abstracted layout of the prototype is shown in Figure 4.23. The capacitive sensors are con-trolled by three OpenCapSense boards; the knock detection is performed on an Arduino Uno microcontroller board. The data fusion is outsourced to a Mini-PC that can be placed in the table. Various evaluations have been performed with the CapTap. We have benchmarked the hand localization against the Leap Motion, concluding that the algorithm works reasonably precise in most parts of the interaction area. The next study was a quantitative study of the percentage of correctly recognized knocks, resulting in considerable misattribution of single and double knocks,

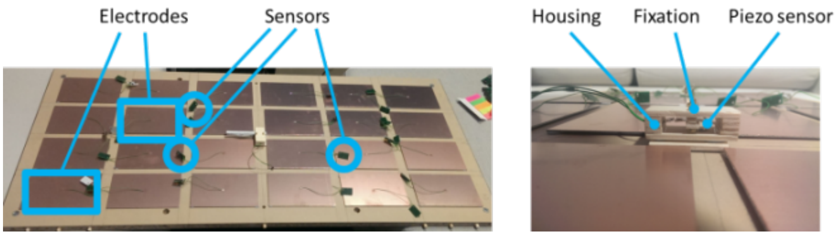

Figure 4.22.: Detail views of the prototype system: left - electrode and sensors, right - knock detection box [BF13]

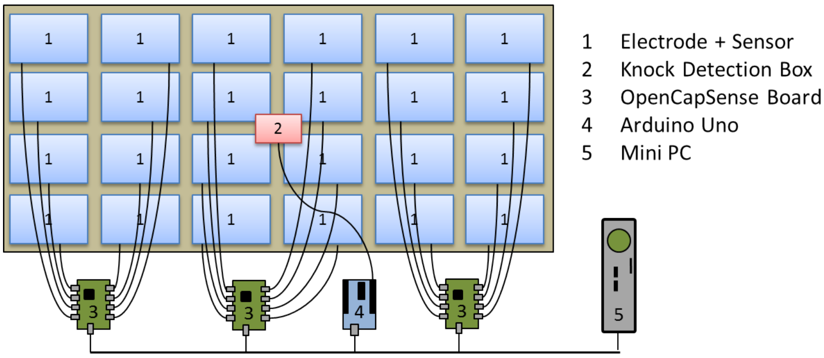

Figure 4.23.: Abstracted view of CapTap prototype including capacitive sensing electrodes and knock detection sensor [BF13]

due to strongly varying knocking styles. However, the presence of any knock was detected with a precision of about 90% [BF13]. Our main evaluation of the system was concerned with the influence of our knock detection on the overall interaction speed of the system. The results concluded that merely adding the knock detection is not enough but that additionally the interfaces have to be adapted towards capacitive systems [BF13].

5. Evaluating capacitive proximity sensors in smart environments

In the previous sections we have presented back-ground information on capacitive proximity sensors and various prototypes of this technology in different application domains within smart environments. In the following section we are building on the collected information to perform a meta-analysis of the acquired data, discussing benefits and limitations of the technology, compare it to competing technologies and give some guidelines to parties interested in developing further applications in this domain.

5.1. Classification of capacitive proximity sensors

Classification

5.2. Comparison to other sensing technologies

In order to properly place capacitive proximity sensing in the smart environment domain it is neces-sary to include a comparison to other sensing tech-nologies. We have chosen systems that have a broad applicability and have been used in various smart environment applications. A short overview can be found in Table 7. We have included a comparison of application domains, environmental influences, de-tection range, processing complexity and unobtru-siveness of the technology. Capacitive touch sensing, as opposed to capacitive proximity sensing relies on an electrode being touched instead of an object being in proximity and is ubiquitous in touch screen applications. RGB cameras are a class of image sensors operating in the same frequency domain as the human eye. They are capable of processing different colors. Infrared cameras operate in near light frequencies that are invisible to the human eye. This allows for application in dark environments and we can project infrared light into the scene without disturbing the user. Ultrasound sensing is using a low frequency range just above the audible limit of human hearing. The waves propagate similar to sound signals and we can perform reflection measurements or time-of-flight methods. Microphone arrays detect signals in the range of human hearing, and thus work with audible signals, such as human speech. Radiofrequency (RF) sensing uses signals in a range between several hundred kHz up to 5GHz, typically used for wireless communication. Commonly the signal strength or time of flight is used to gather information about the environment. Most technologies are capable of supporting multiple application domains. Some non-intuitive examples include WiSee that enables whole-body gestural interaction using WiFi signals [86] or MoGees that uses a single microphone to enable gesture interfaces on various surfaces [87]. Capacitive sensors are disturbed by conductive objects and electric fields, whereas cameras struggle with occlusion and additional light sources. Occlusion is a weak point, and a line of sight is required. Sound sensors are prone to dampening materials and environmental noise interfering with the signal. RF signals usually propagate well through most materials and only external sources may be an issue. The detection range of the technologies varies strongly. RF ranges before light, sound and electric fields. However, this again strongly depends on apxplication and layout of the sensing devices. It is not easy to find

Table 5.1.: Add caption

Name	Application Domains	Environmental Influences	Detection Range	Processing Complexity	Unobtrusiveness
Capacitive proximity sensing	indoor localization, smart appliances, physiological sensing, gestural interaction	electric fields, conductive objects	near distance (< 100cm)	Few high dynamic range data sources	invisible integration possible
Capacitive touch sensing	smart appliances, physiological sensing, gestural interaction	electric fields, conductive objects	touch	Few binary sensors	thin cover above electrodes
RGB cameras	indoor localization, smart appliances, physiological sensing, gestural interaction	occlusion, external lights	far distance (> 10m)	Complex image processing based on resolution	pinhole lenses
Infrared cameras	indoor localization, physiological sensing, gestural interaction	occlusion, external infrared light	medium distance (< 5m)	Complex image processing based on resolution	infrared source and camera
Ultrasound sensing	indoor localization, smart appliances, gestural interaction	acoustic occlusion, absorbing materials	medium distance (< 5m)	Few low dynamic range data sources	emitter and senders with exposed pinhole speaker, microphone
Microphone arrays	indoor localization, smart appliances, physiological sensing	environmental noise, absorbing materials	medium distance (< 5m)	Very high dynamic range data sources	exposed pinhole microphones
Radiofrequency sensing	indoor localization, smart appliances, gestural interaction	other RF devices	far distance (> 10m)	Few low dynamic range data sources	hidden emitters and senders possible

a good measure about the processing complexity associated to a different sensing technology. We are using a simplified model, taking the dynamic range of a sensor and the number of sensors typically required. Dynamic range is the difference between the smallest detectable value and the largest detectable value. Microphones have a high dynamic range measuring over a larger frequency scale, whereas touch sensors only have two different states. Finally capacitive sensors and RF sensors can be applied completely invisible. Cameras, microphones and ultrasound need a direct connection to the out-side world. However, there are very small variants available that are barely visible to the naked eye.

5.3. Limitations

Table 5.2.: Overview of capacitive proximity sensing limitations

Name	Examples
Environmental influence	Static electric fields, dynamic electric fields, temperature, humidity, conductive objects
Physical range	Small differences in capacitance, reduction due to influences, physical limitations
Object detection	Small number of data points, a priori knowledge

Despite the potential that has been described in the previous sections there are various limitations of capacitive proximity sensing that we can put into the different groups of environmental influence, physical range and object detection that will be described in more detail in the following section. A short overview is given in Table 5.2.

5.3.1. Environmental Influence

One of the main limitations of capacitive proximity sensors is their sensitivity towards environmental influences. Any factor that modifies an electric field will also affect the measurement of a capacitive sensor. The current environmental parameters, like temperature and humidity are having a considerable effect on the atmosphere in which the electric field propagates. However, those changes are usually over a longer period of time and can be compensated using a factor for drift, as described in the previous sections about noise reduction. A more challenging factor is the other electric devices in the environment that emit stronger electromagnetic fields. While persistent sources, such as permanent electric installations can usually be countered using a galvanic isolation there are other non-obvious challenges. E.g. we noticed that certain plasma TVs are able to disturb the measurement and increase noise levels consider-ably. This change is even varying according to screen content. A minor effect is the presence of high-frequency fields that are getting more prevalent in modern IT equipped environments. Instead of the 2.4GHz and 5GHz ranges that are often used in wire-less communication, capacitive proximity sensors can operate in the range of a few kHz to one MHz. An additional issue might arise when placing sensors close to each other. The created electric fields may disturb the measurement if some electrodes are charged and create fields to adjacent electrodes while they are discharged for measurement. Consequently, specific charge-discharge cycles or multiplexing methods have to be used to counter this effect. A major challenge is dealing with conductive objects that are permanently placed in the immediate sensing environment. It is difficult to distinguish the object we want to detect from a disturbing object, if their influence on the electric field is similar. Long term data analysis may help in performing a successful detection. The CapFloor prototype is affected by environmental influences the most, given the small size of the electrodes relative to the interaction area and the changing environment on top of the floor. We are using a strong noise reduction algorithm and drift compensation to create a more stable result while reducing the detection range.

5.3.2. Physical Range

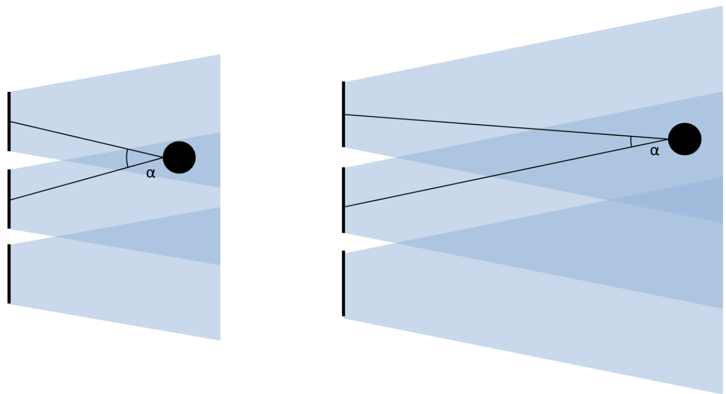

Figure 5.1.: Reduced angular resolution on smaller, distant objects

The physical range of the generated electric field is one of the main limiting factors of capacitive proximity sensing. In order to detect objects that are further away we have to increase the electric field strength sufficiently. This is easier the larger the electrode is, as its potential capacitance is higher. However, this also leads to distant objects having an ever smaller influence on the overall capacitance, and we need more precise measurement circuits and longer measurement times to improve the signal-to-noise ratio. Additionally, looking at smaller objects the angular resolution will decrease as shown in Figure 5.1. This makes it more difficult to get a precise localization as the immanent noise leads to an angular error. While this can be compensated using more sensors, the far distance would require us to use large electrodes that have to be placed further apart resulting in a huge area that would have to be equipped with sensor electrodes. In general the achievable resolution is not comparable to vision based system and has to be taken into consideration when designing the specific application. A balance between electrode size, physical range and achievable resolution has to be found. The MagicBox size does not allow an integration of very large electrodes. Instead we are optimizing the available space in order to achieve a detection that lets us detect hands in a distance between 15 and 20 centimeters.

5.3.3. Object Detection

Object detection using capacitive sensors can be partially compared to object detection using camera systems, with a single sensor being equivalent to a single photo sensor. The light intensity measure is comparable to field intensity and likewise we can't distinguish if the measurement is caused by a weak source in close proximity or a strong source at a further distance. As a practical example the capacitive sensor can't decide if one hand is close to the sensor or two hands are a bit further away. This effect makes it challenging to provide object detection and we usually have to combine the information from various sensors to get a good idea about object shape and size. Due to the presented challenges in physical range and electrode size, capacitive proximity sensing systems do

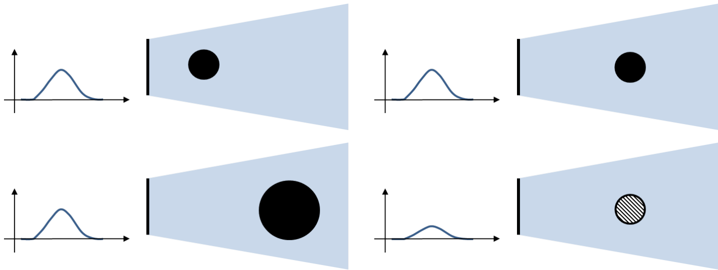

Figure 5.2.: Same response to differently sized objects (left), different response to varying materials (right)

not have the same level of scalability as opposed to cameras, where millions of photo sensors can be placed in very small areas. Additionally, the effect of an object on the electric field is not always closely correlated to the object dimensions, but instead based on conductivity, material and other factors. We may get the same response to different objects at different distances or get a varying on similarly sized objects made of different materials, as shown in Figure 5.2. The Active Armrest has gestures for one and two fingers that are distinguished using a simple threshold. If another object is entering the field or the person has larger fingers the system will fail to properly differentiate gestures. Accordingly some other compensation methods should be used.

5.4. Benefits

Table 5.3.: Overview of capacitive proximity sensing benefits

Name	Examples
Versatility	Flexible electrode design, scalability, different sensing methods
Unobtrusiveness	Invisible application, non-disturbing frequency range
Processing Complexity	Small number of sensors, variable dynamic range

After discussing the various limitations of capacitive proximity sensing, the following section will give an overview of the benefits. Similar to the previous section we have three groups, namely versatility, unobtrusiveness and processing complexity. Some examples within these groups are shown in Table 6.

5.4.1. Versatility

A main benefit of capacitive proximity sensing is the versatility in which they can be applied. The flexibility of electrode materials, size and geometry allows specifically creating highly individual applications. Example electrodes include transparent metal oxide layers, woven conductive thread, copper wires, PCB boards or simple aluminum foil. The sensors systems are also highly scalable. By choosing appropriate voltages and frequencies it is possible to add a high number of sensors to a single object. Using smart measurement windows and different multiplexing methods, sensors can be placed close together and electrodes may act as both sender and receiver. The different sensing methods presented - loading mode, shunt mode and transmit mode enable a variety of different sensing patterns. The human body can be used as both sender and receiver and smart electrode layouts allow using a smaller number of processing units. In conclusion, it is possible to add capacitive sensing to most everyday objects to enable different forms of interaction, create natural interfaces and smart objects. Our prototypes are using different electrode materials, flexible or solid electrodes, conductive thread, wires, shielded or non-shielded layouts.

5.4.2. Unobtrusiveness

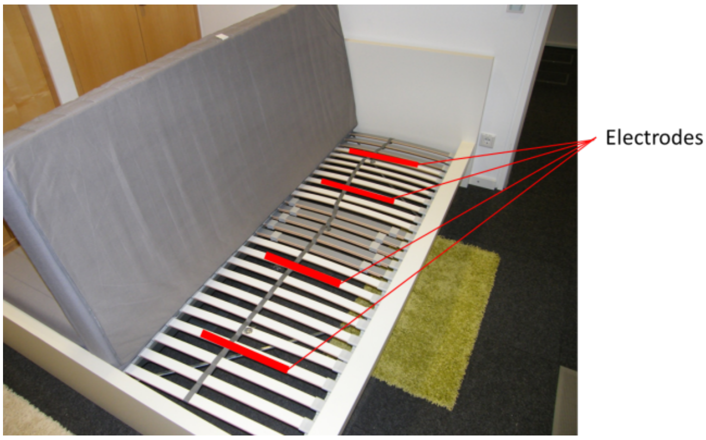

Figure 5.3.: Electrodes and sensors hidden below mattress of Smart Bed

Electric fields are not usually perceived by persons, unless they are of exceptional strength. Furthermore they propagate through many materials that we are typically using in our environment, including most plastics, wood or tiles. This allows us to invisibly apply capacitive proximity sensors without a strong effect on the measurement. Application below several centimeters of covering is possible, if the electrodes are designed properly for sensing in this distance. The frequency range in which the sensors are operating is usually not in an interval that disturbs other electronic systems. Thus it is feasible to use capacitive sensing even in environments, where non-disturbance is a main requirement. Additionally the used frequencies are not considered to be biologically active, and good results can be achieved using small currents. It is possible to equip most conductive objects directly with capacitive proximity sensors and hide them below non-conductive objects with minimal spatial requirements. Our Smart Bed and Active Armrest prototypes are using sensor sets that are completely invisible from the outside and communicate wirelessly to a PC only using a power supply. Figure 5.3 shows the electrodes and sensors hidden below the mattress of the Smart Bed.

5.4.3. Processing Complexity

An appropriate analogy of capacitive proximity sensors is a single photodiode. As opposed to a light intensity we are measuring capacitance. While the information we can gain from such a measurement is limited, the processing required to analyze the signal is also low. Performing signal analysis on an array of 16 capacitive sensors is comparable to processing the image of a 4x4 pixel camera. Therefor it is easy to create highly integrated systems with very low-power devices for performing any subsequent data analysis. While it is possible and in many cases beneficial to use complex data processing algorithms for object detection it is in most cases still possible to replace them with simpler methods for a comparable result. In many applications it is even viable to opt for a quantized capacitance measurement. In the case of a touch sensor a single binary measure is sufficient. However, it is also possible to select various different levels and reduce the dynamic range to an easily computable value that is 4 or 8 Bit long. Depending on the chosen algorithm this dynamic range reduction can occur either in pre-processing or high level processing. With the exception of the Capacitive Chair our prototypes are using simple data processing methods that can be easily applied on embedded systems. A preferred method for object localization is the weighted average algorithm. Regarding model-based data processing, even very simple cylindrical models, such as the one used for the Smart Bed, are capable to reliably predict numerous postures that are relevant in real world applications. In general, the low requirements for data preprocessing, allows dedicating more resources to high level data processing algorithms if the specific application is resource con-strained. The OpenCapSense toolkit that is the base for most of our prototypes has a fairly powerful micro-controller that is able to implement all of the processing steps - thus enabling highly integrated, low-power capacitive proximity sensing prototypes that can be used in smart environment applications.

5.5. Guidelines

After discussing the limitations and benefits of capacitive proximity sensors, the final section of this chapter will give some general guidelines on their application. The first step of this process is a decision if capacitive sensors technology is suitable for the given application. This part should be driven by three questions. What do I need to measure in my application scenario? Capacitive proximity sensors can measure the presence and properties of conductive, grounded objects. This includes the various application scenarios shown in the previous sections. However, if the application requires measuring properties of unsupported objects that are non-conductive, a different technology should be chosen.

What sensing technologies are supporting the required measurements?

It may be the case that multiple technologies sup-port the measurements required in this specific applications. Cameras often can provide similar recognition as capacitive sensors, e.g. in indoor localization applications. In this step all potential sensing technologies should be collected. Are capacitive proximity sensors beneficial for my scenario? An evaluation of the different candidates is the final step and should lead to a decision about the most suitable sensing technologies. If the distance is too high for capacitive proximity sensors or enough processing power is available and lighting conditions are static, cameras might be more suitable. This should be driven by the different benefits and limitations of the technologies. If there is a decision in favor of capacitive sensors the next step is to design the specific electrode layout. Similar to technology selection we can use a few basic questions to get an idea of what layout to use.

How many sensors are required to get the measurement?

The number of sensors required is depending on the area we want to cover, the specific object parameters that have to be determined and the desired resolution. The electrodes are inherently limited in size, as a single sensor can only charge and discharge to a specific maximum capacity. Therefore, if a large area has to be covered more electrodes and sensors are necessary. If we just want to measure the presence of a hand a single electrode may suffice. If orientation and position are interesting we need to combine measurements from various sensors.

What should be the size and geometry of the electrodes?

This is closely related to the previous question. If the application is not restricting the available space, the electrode should be approximately of the same size as the object that is to be detected. This generates the highest difference in capacitance when the distance is changing.

What is the best electrode material to use?

Copper is always a good first choice to create electrodes. If elasticity is necessary we can use copper foil and solid copper if that is of no concern. For transparent electrodes we will have to use one of the previously presented materials, such as ITO. If electrodes have to be integrated into cloth, conductive thread is a good candidate. Any conductive material will act as an electrode, thus the application and budget should be the primary driver of this decision.

Does my application require any shielding?

Shielding allows detecting only objects approaching from a certain direction. If the application re-quires this additional hardware, because it is anticipated that other objects might disturb the measurement, shielding should be used. Finally, if the hardware is designed as desired the different variations of data processing have to be chosen and configured according to the application. Using baseline calibration is beneficial in the vast majority of applications. Having a distinct starting point simplifies all further steps of high-level data processing, such as normalization and setting different thresholds. This step may only be omitted in very stable environments and if the system has sufficient a priori information to operate on raw data. Drift compensation should be handled in a similar fashion. The common methods are not computationally expensive and having a stable baseline over time allows the same algorithms to be applied in a more robust fashion. The method and configuration of noise reduction are strongly depending on the specific case. Some form of noise reduction might be required in most applications. Yet, according to the type of noise different methods can be used. If outliers are an issue a median filter is appropriate; if a smoother signal is desired an average filter can be used. Regarding high-level data processing there are manifold variations of methods. Data-driven machine learning algorithms are a good method if we have a small set of potential outcomes of our applications, e.g. the different postures that could be recognized on a chair or couch. If our application has many different potential outcomes, e.g. the thousands of potential locations in a hand tracking system, it is typically beneficial to use a model-driven approach. However, these models may be supported by data-driven algorithms, such as particle filters. One example is the Swiss-Cheese object tracker by Grosse-Puppendahl et al. [GPBKK13]. The data processing examples shown in the previous sections give an idea of the decision rationale in various application domains. We can say in conclusion that capacitive proximity sensors are a viable, or even, ideal solution for a considerable number of different applications. However, a certain level of preparation is required in the design process to create a system that benefits from the technology.

6. AmbiTrack indoor localization system for smart environments

6.1. Background

6.1.1. Technologies

6.1.2. Smart environment applications and requirements

6.2. AmbiTrack system

6.2.1. EvAAL

6.2.2. System Design

6.2.3. Prototype

6.2.4. Evaluation

7. Conclusions and Future Work

This chapter summarizes what a great job you did and what could be done if you could do as a second PHD thesis :)

Acknowledgments

While many consider writing a PhD to be a mostly personal endeavor there are always various sources of discourse, collaboration, support and inspiration. So in no particular order there are various persons or groups of persons that deserve credit:

A. Publications and Talks

The thesis is partially based on the following publications and talks:

A.1. Publications

1. publication 1
2. publication 2
3. ....
4. publication x

A.2. Talks

1. talk 1
2. ...
3. talk n

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. Große-Puppendahl, Tobias - Multi-hand Interaction Using Custom Capacitive Proximity Sensors - MSc TU Darmstadt 2012
2. Berghöfer, Yannick - Human-Machine-Interfaces in Automotive Environments using Capacitive Proximity Sensors - MSc TU Darmstadt 2013
3. Krepp, Stefan - Unobtrusive Surface Touch Recognition using Acoustic Tracking - MSc TU Darmstadt 2014

B.2. Bachelor Thesis

1. Fischer, Arthur - Unterstützung von zielbasierter Interaktion durch gestenerkennende Zeigegeräte - BSc TU Darmstadt 2012
2. Majewski, Martin - Visual-aided Selection of Reactive Elements in Intelligent Environments - BSc TU Darmstadt 2012
3. Neumann, Stephan - Automotive interfaces using an interactive armrest - BSc TU Darmstadt 2014

C. Curriculum Vitae

Personal Data

Name	Andreas Braun
Birth date & place	04.10.1982 Aschaffenburg, Germany
Nationality	German

Education

2008 – 2010	Master of Science in Computational Engineering at Technical University of Darmstadt, Germany
2004 – 2008	Bachelor of Science in Computational Engineering at Technical University of Darmstadt, Germany
2002 – 2004	Study of Physics at Julius-Maximilians Universität in Würzburg, Germany

Work Experience

2010 –	Researcher, Competence Center Interactive Multimedia Appliances, Fraunhofer Institute for Computer Graphics Research, Darmstadt, Germany, Focus: HCI applications in smart environments
2008 – 2009	Student Assistant, Competence Center Interactive Multimedia Appliances, Fraunhofer Institute for Computer Graphics Research, Darmstadt, Germany, Focus: Sensor applications and interactive systems

Bibliography

[AKKM13] ADIB F., KABELAC Z., KATABI D., MILLER R. C.: 3D Tracking via Body Radio Reflections. xi, 28

[Bax96] BAXTER L. K.: Capacitive Sensors. *Sensors Peterborough NH*, 1 (1996), 1–17. 7, 16, 17

[BB98] BERTOZZI M., BROGGI A.: GOLD: A parallel real-time stereo vision system for generic obstacle and lane detection. *Image Processing, IEEE Transactions on 7*, 1 (1998), 62–81. 40

[BDK13] BRAUN A., DUTZ T., KAMIETH F.: Capacitive sensor-based hand gesture recognition in ambient intelligence scenarios. In *Proceedings of the 6th International Conference on PErvasive Technologies Related to Assistive Environments* (2013), ACM, p. 5. 50, 52

[BF13] BRAUN A., FRANK S.: The Capacitive Chair. 2013. xii, 46, 47, 53, 54, 55

[BH11] BRAUN A., HAMISU P.: Designing a multi-purpose capacitive proximity sensing input device. In *Proceedings of the 4th International Conference on PErvasive Technologies Related to Assistive Environments PETRA 11* (2011), ACM Press. xii, 50, 51, 52, 53

[BH12] BRAUN A., HEGGEN H.: Context recognition using capacitive sensor arrays in beds. In *Technik für ein selbstbestimmtes Leben - 5. Deutscher AAL-Kongress* (Berlin, 2012), VDE VERLAG GmbH. xii, 43, 44, 45

[BHK97] BELHUMEUR P. N., HESPANHA J. A. P., KRIEGMAN D. J.: Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 19*, 7 (1997), 711–720. 17

[BHW12] BRAUN A., HEGGEN H., WICHERT R.: CapFloor – A Flexible Capacitive Indoor Localization System. In *Evaluating AAL Systems Through Competitive Benchmarking. Indoor Localization and Tracking* (2012), Chessa S., Knauth S., (Eds.), Communications in Computer and Information Science, pp. 26–35. xi, 41, 43

[BKKF13] BECK S., KUNERT A., KULIK A., FROEHLICH B.: Immersive Group-to-Group Telepresence. *Visualization and Computer Graphics, IEEE Transactions on 19*, 4 (2013), 616–625. 23

[BO10a] BARRETT G., OMOTE R.: Projected capacitive touch screens. *Inf. Disp. Mag* (2010). xi, 11

[BO10b] BARRETT G., OMOTE R.: Projected-capacitive touch technology. *Information Display* (2010). 9, 10

[CC89] CAMP R. C., CAMP ROBERT C.: Benchmarking: the search for industry best practices that lead to superior performance. 31

[CD07] COOK D. J., DAS S. K.: How smart are our environments? An updated look at the state of the art. *Pervasive and mobile computing 3*, 2 (2007), 53–73. xiii, 2, 19

[CK12] CHESSA S., KNAUTH S.: *Evaluating AAL Systems Through Competitive Benchmarking. Indoor Localization and Tracking*, vol. 309 of *Communications in Computer and Information Science*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012. 3, 32, 42

[Cla62] CLARKE A. C.: Hazards of prophecy: The failure of imagination. *Profiles of the Future, Gollancz, London* (1962). 1

[Com06] COMMONS W.: A bayer pattern on a sensor in isometric perspective/projection, 2006. xi, 20

[Cor13] CORPORATION C. S.: CY3271 PSoC® FirstTouchTM Starter Kit with CYFiTM Low-Power RF, 2013. 51

[Cro09] CROLOTTE A.: Issues in benchmark metric selection. In *Performance Evaluation and Benchmarking*. Springer, 2009, pp. 146–152. 32

[CRV08] CORBISHLEY P., RODRÍGUEZ-VILLEGAS E.: Breathing detection: towards a miniaturized, wearable, battery-operated monitoring system. *Biomedical Engineering, IEEE Transactions on 55*, 1 (2008), 196–204. 26

[Cyp12] CYPRESS: *Cypress TrueTouch® Touchscreen Solution Drives "Floating Touch" Navigation Feature in New Xperia^{TM} sola Smartphone from Sony Mobile Communications*. Tech. rep., 2012. 9

[DBM13] DJAKOW M., BRAUN A., MARINC A.: MoviBed - sleep analysis using capacitive sensors. 2013. xii, 43, 44, 45, 46

[DL01] DIETZ P., LEIGH D.: DiamondTouch. In *Proceedings of the 14th annual ACM symposium on User interface software and technology - UIST '01* (New York, New York, USA, Nov. 2001), ACM Press, p. 219. 11

[DLP03] DONGARRA J. J., LUSZCZEK P., PETITET A.: The LINPACK benchmark: past, present and future. *Concurrency and Computation: practice and experience 15*, 9 (2003), 803–820. 31

[Fra13] FRANK R.: *Understanding smart sensors*. Artech House, 2013. 19

[GePMB11] GROSS E PUPPENDAHL T., MARINC A., BRAUN A.: Classification of User Postures with Capacitive Proximity Sensors in AAL-Environments. *Ambient Intelligence 7040* (2011), 314–323. xi, 12

[Gli00] GLINSKY A.: *Theremin: Ether music and espionage*. University of Illinois Press, 2000. xi, 18

[GPBB*13] GROSSE-PUPPENDAHL T., BERGHOEFER Y., BRAUN A., WIMMER R., KUIJPER A.: OpenCapSense: A Rapid Prototyping Toolkit for Pervasive Interaction Using Capacitive Sensing. *IEEE International Conference on Pervasive Computing and Communications (PerCom) 18* (2013), 22. xi, 10, 11

[GPBKK13] GROSSE-PUPPENDAHL T., BRAUN A., KAMIETH F., KUIJPER A.: Swiss-cheese extended: an object recognition method for ubiquitous interfaces based on capacitive proximity sensing. In *Proceedings of the 2013 ACM annual conference on Human factors in computing systems* (2013), ACM, pp. 1401–1410. 16, 64

[GZBB09] GEORGE B., ZANGL H., BRETTERKLIEBER T., BRASSEUR G.: Seat occupancy detection based on capacitive sensing. *Instrumentation and Measurement, IEEE Transactions on 58*, 5 (2009), 1487–1494. 17

[HB10] HAMISU P., BRAUN A.: Analyse des Schlafverhaltens durch kapazitive Sensorarrays zur Ermittlung der Wirbelsäulenbelastung. In *3. Deutscher AAL Kongress* (2010), VDE VERLAG GmbH, pp. 3–6. 43

[Hen00] HENNING J. L.: SPEC CPU2000: Measuring CPU performance in the new millennium. *Computer 33*, 7 (2000), 28–35. 31

[HPR00] HAN J., PRATT M., REGLI W. C.: Manufacturing feature recognition from solid models: a status report. *Robotics and Automation, IEEE Transactions on 16*, 6 (2000), 782–796. 17

[HSH11] HARRISON C., SCHWARZ J., HUDSON S. E.: TapSense: enhancing finger interaction on touch surfaces. In *Proceedings of the 24th annual ACM symposium on User interface software and* *technology* (2011), ACM, pp. 627–636. xi, 25, 26

[HW08] HAKLAY M., WEBER P.: Openstreetmap: User-generated street maps. *Pervasive Computing, IEEE 7*, 4 (2008), 12–18. 49

[iee07] IEEE Standard for a Smart Transducer Interface for Sensors and Actuators Wireless Communication Protocols and Transducer Electronic Data Sheet (TEDS) Formats, 2007. 19

[IHQ04] INGRAM S. J., HARMER D., QUINLAN M.: Ultrawideband indoor positioning systems and their use in emergencies. In *Position Location and Navigation Symposium, 2004. PLANS 2004* (2004), IEEE, pp. 706–715. 3

[IKH11] IZADI S., KIM D., HILLIGES O.: KinectFusion: real-time 3D reconstruction and interaction using a moving depth camera. *Proceedings of the 24th ...* (2011), 559 – 568. xi, 22, 23

[Inc13] INC. S.: Global Posture Study, 2013. 46

[Jai91] JAIN R.: *The art of computer systems performance analysis*, vol. 182. John Wiley & Sons Chichester, 1991. 32

[KBBC05] KALTENBRUNNER M., BOVERMANN T., BENCINA R., COSTANZA E.: TUIO: A protocol for table-top tangible user interfaces. In *Proc. of the The 6th Int'l Workshop on Gesture in Human-Computer Interaction and Simulation* (2005). 49

[KJJ11] KREJCAR O., JIRKA J., JANCKULIK D.: Use of mobile phones as intelligent sensors for sound input analysis and sleep state detection. *Sensors* (2011). 43

[LBO11] LESER R., BACA A., OGRIS G.: Local positioning systems in (game) sports. *Sensors 11*, 10 (2011), 9778–9797. 3

[LC85] LEWIS B. C., CREWS A. E.: The evolution of benchmarking as a computer performance evaluation technique. *MIS Quarterly* (1985), 7–16. 31

[Lew04] LEWIS F. L.: Wireless sensor networks. *Smart environments: technologies, protocols, and applications* (2004), 11–46. 19

[LWG05] LANDSIEDEL O., WEHRLE K., GÖTZ S.: Accurate prediction of power consumption in sensor networks. In *Proc. of the Second Workshop on Embedded Networked Sensors* (2005), Citeseer. 33

[Mic] MICROSOFT: What Is the Windows Experience Index. 31

[Mit07] MITCHELL H. B.: *Multi-sensor data fusion*. Springer, 2007. 15

[MPLK05] MOON J., PARK J., LEE T., KIM Y.: Transparent conductive film based on carbon nanotubes and PEDOT composites. *Diamond and related materials 14*, 11 (2005), 1882–1887. 10

[Nok12] NOKIA: *Puremotion Technology White Paper*. Tech. rep., 2012. 9

[OHR13] OCHIAI Y., HOSHI T., REKIMOTO J.: Three-dimensional Mid-air Acoustic Manipulation by Ultrasonic Phased Arrays. *arXiv preprint arXiv:1312.4006* (2013). xi, 24, 25

[Pan12] PANGER G.: Kinect in the kitchen: testing depth camera interactions in practical home environments. In *CHI'12 Extended Abstracts on Human Factors in Computing Systems* (2012), ACM, pp. 1985–1990. xi, 22, 23

[PC00] PENTLAND A., CHOUDHURY T.: Face recognition for smart environments. *Computer 33*, 2 (2000), 50–55. xi, 20, 21

[PCB00] PRIYANTHA N. B., CHAKRABORTY A., BALAKRISHNAN H.: The cricket location-support system. In *Proceedings of the 6th annual international conference on Mobile computing and networking* (2000), ACM, pp. 32–43. xi, 24

[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* (2013), ACM, pp. 27–38. xi, 28, 29

[PJS*13] PHAM C., JACKSON D., SCHÖNING J., BARTINDALE T., PLÖTZ T., OLIVIER P.: FoodBoard: surface contact imaging for food recognition. In *Proceedings of the 2013 ACM international joint conference on Pervasive and ubiquitous computing* (2013), ACM, pp. 749–752. xi, 21, 22

[Pos11] POSLAD S.: *Ubiquitous computing: smart devices, environments and interactions*. Wiley. com, 2011. 1, 2

[PSF01] PRASSLER E., SCHOLZ J., FIORINI P.: A robotics wheelchair for crowded public environment. *Robotics & Automation Magazine, IEEE 8*, 1 (2001), 38–45. xi, 24

[RAMB*05] RANGANATHAN A., AL-MUHTADI J., BIEHL J., ZIEBART B., CAMPBELL R. H., BAILEY B.: Towards a pervasive computing benchmark. In *Pervasive Computing and Communications Workshops, 2005. PerCom 2005 Workshops. Third IEEE International Conference on* (2005), IEEE, pp. 194–198. 31, 32

[Rek02] REKIMOTO J.: SmartSkin: an infrastructure for freehand manipulation on interactive surfaces. In *Proceedings of the SIGCHI conference on Human factors in computing systems: Changing our world, changing ourselves* (2002), ACM, pp. 113–120. 3

[SAW94] SCHILIT B., ADAMS N., WANT R.: Context-aware computing applications. In *Mobile Computing Systems and Applications, 1994. WMCSA 1994. First Workshop on* (1994), IEEE, pp. 85–90. 1

[SDKS10] SCHMIDT A., DEY A. K., KUN A. L., SPIESSL W.: Automotive user interfaces: human computer interaction in the car. In *CHI'10 Extended Abstracts on Human Factors in Computing Systems* (2010), ACM, pp. 3177–3180. 48

[SFC*13] 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. *Communications of the ACM 56*, 1 (June 2013), 116–124. 22

[SKOM06] SUGANO M., KAWAZOE T., OHTA Y., MURATA M.: Indoor localization system using RSSI measurement of wireless sensor network based on ZigBee standard. *Target 538* (2006), 50. xi, 27

[SL86] SALMI T., LEINONEN L.: Automatic analysis of sleep records with static charge sensitive bed. *Electroencephalography and Clinical Neurophysiology 64*, 1 (1986), 84–87. 44

[Smi88] SMITH J. E.: Characterizing computer performance with a single number. *Communications of the ACM 31*, 10 (1988), 1202–1206. 32

[Smi96] SMITH J. R.: Field mice: Extracting hand geometry from electric field measurements. *IBM Systems Journal 35*, 3.4 (1996), 587–608. xi, 3, 10

[Smi99] SMITH J. R.: *Electric field imaging*. PhD thesis, Massachusetts Institute of Technology, 1999. 9, 16, 18

[SPSS11] SUNG J., PONCE C., SELMAN B., SAXENA A.: Human Activity Detection from RGBD Images. In *Plan, Activity, and Intent Recognition* (2011). 22

[TCD*00] THOMAS B., CLOSE B., DONOGHUE J., SQUIRES J., DE BONDI P., MORRIS M., PIEKARSKI W.: ARQuake: An outdoor/indoor augmented reality first person application. In *Wearable Computers, The Fourth International Symposium on* (2000), IEEE, pp. 139–146. 3

[TKA06] TABAR A. M., KESHAVARZ A., AGHAJAN H.: Smart home care network using sensor fusion and distributed vision-based reasoning. In *Proceedings of the 4th ACM international workshop on Video surveillance and sensor networks* (2006), ACM, pp. 145–154. xi, 20, 21

[TSPM01] TAN H. Z., SLIVOVSKY L. A., PENTLAND A., MEMBER S.: A sensing chair using pressure distribution sensors. *Mechatronics, IEEE/ASME Transactions on 6*, 3 (2001), 261–268. 46

[Val08] VALLI A.: The design of natural interaction. *Multimedia Tools and Applications 38*, 3 (2008), 295–305. 2

[VMV09] VALTONEN M., MAENTAUSTA J., VANHALA J.: TileTrack: Capacitive human tracking using floor tiles. In *IEEE International Conference on Pervasive Computing and Communications* (Mar. 2009), IEEE, pp. 1–10. 12

[WB95] WELCH G., BISHOP G.: An introduction to the Kalman filter, 1995. 15

[Wei84] WEICKER R. P.: Dhrystone: a synthetic systems programming benchmark. *Communications of the ACM 27*, 10 (1984), 1013–1030. 31

[Wei91] WEISER M.: The Computer for the 21st Century. *Scientific American 265*, 3 (1991), 94–104. xi, 1, 21, 23

[WF07] WILSON T. V., FENLON W.: How the iPhone Works - Multi-touch systems, 2007. 17

[WHKS06] WIMMER R., HOLLEIS P., KRANZ M., SCHMIDT A.: Thracker - Using Capacitive Sensing for Gesture Recognition. In *26th IEEE International Conference on Distributed Computing Systems Workshops (ICDCSW'06)* (2006), IEEE, pp. 64–64. 17

[Wil04] WILSON J. S.: *Sensor technology handbook*. Access Online via Elsevier, 2004. 32

[WP10] WILSON J., PATWARI N.: Radio tomographic imaging with wireless networks. *Mobile Computing, IEEE Transactions on 9*, 5 (2010), 621–632. xi, 27, 28

[WRS*12] WU H.-Y., RUBINSTEIN M., SHIH E., GUTTAG J., DURAND F., FREEMAN W.: Eulerian video magnification for revealing subtle changes in the world. *ACM Transactions on Graphics 31*, 4 (July 2012), 1–8. xi, 20, 22

[WTT13] WATANABE H., TERADA T., TSUKAMOTO M.: Ultrasound-based movement sensing, gesture-, and context-recognition. In *Proceedings of the 17th annual international symposium on International symposium on wearable computers* (2013), ACM, pp. 57–64. xi, 24

[WYG*09] WRIGHT J., YANG A. Y., GANESH A., SASTRY S. S., MA Y.: Robust face recognition via sparse representation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 31*, 2 (2009), 210–227. 20

[XLL*13] XU C., LI S., LIU G., ZHANG Y., MILUZZO E., CHEN Y.-F., LI J., FIRNER B.: Crowd++: unsupervised speaker count with smartphones. In *Proceedings of the 2013 ACM international joint conference on Pervasive and ubiquitous computing* (2013), ACM, pp. 43–52. xi, 25, 26

[YJS06] YILMAZ A., JAVED O., SHAH M.: Object tracking: A survey. *Acm Computing Surveys (CSUR) 38*, 4 (2006), 13. 16

[ZFBZ08] ZHANG C., FLORÊNCIO D., BA D. E., ZHANG Z.: Maximum likelihood sound source localization and beamforming for directional microphone arrays in distributed meetings. *Multimedia, IEEE Transactions on 10*, 3 (2008), 538–548. xi, 25, 26

[Zha12] ZHANG Z.: Microsoft kinect sensor and its effect. *Multimedia, IEEE 19*, 2 (2012), 4–10. xi, 21, 23

[ZSP*95] ZIMMERMAN T. G., SMITH J. R., PARADISO J. A., ALLPORT D., GERSHENFELD N.: Applying electric field sensing to human-computer interfaces. In *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '95* (New York, New York, USA, 1995), no. May, ACM Press, pp. 280–287. 18