From the Institute of Telematics
of the University of Luebeck
Director: Prof. Dr. S. Fischer

Ambient Assisted Living with Dynamic Interaction Ensembles

Dissertation
for Fulfillment of
Requirements
for the Doctoral Degree
of the University of Lübeck

from the Department of Computer Sciences/Engineering

Submitted by

Bashar Altakrouri

from Hebron, Palestine.

Lübeck, August 31, 2014

1. First referee: Prof. Dr.-Ing. Andreas Schrader
2. Second referee: Prof. Dr.-Ing. Nicole Jochems

Date of oral examination August 22, 2014

Approved for printing. Lübeck, August 25, 2014

I lovingly dedicate this thesis to my wife and my parents who supported me each step of the way.

Acknowledgement

It would not have been possible to complete this doctoral thesis without the support of the kind people around me, to only some of whom it is possible to give particular mention here. I would like to thank my principal supervisor, Prof. Dr.-Ing. Andreas Schrader, for his true sense of patience, honesty, and support. In combination with my second supervisor, Prof. Dr. med. Jens-Martin Träder for his insightful advice. I am very blessed to work with intelligent, dynamic, and positively provocative and engaged colleagues Dr. Fahim Kawsar, Dr.-Ing. Darren Carlson, and Lukas Ruge. Moreover, special thanks to the students that I have worked with for their effort and hard work. Special thanks to my colleagues at the Institute of Telematics and the Graduate School for Computing in Medicine and Life Sciences for their continuous support and the fantastic friendly environment.

My wife Samah has been, always, my strengths and my guiding light, and I thank her for reorienting my life. My son Mounir and my daughter Maryam have been inspiring candles to light a world full of happiness and joy.

My parents have been a constant source of support (emotional and moral) during all those years, and this thesis would certainly not have existed without them. They are the first to teach me the nobel and exciting meanings of asking questions and searching for answers, the first to teach me what patience really means, to them this thesis is dedicated. To my mother in law for the unprecedented support I also dedicate this thesis. Moreover, to my lovely brothers, sister, brother-in-law, and sisters-in-law, I am very grateful for having you along side me in this long journey.

Last, but by no means least, I thank my friends in Germany, Great Britain, Palestine and elsewhere for their support and encouragement throughout, only some I can mention here Michal Janiszewski, Fadi Chehimi, Fahd Saleh, Alaa Tamimi, Yousef Salah, Isaac Jwailes, Ibrahim Efe, and Fayez Almari.

For any errors or inadequacies that may remain in this work, the responsibility is entirely my own.

Erklärung

Ich versichere, die vorliegende Arbeit selbstständig und nur unter Benutzung der angegebenen Hilfsmittel angefertigt zu haben.

Lübeck, den August 31, 2014

Preface

The used latex template is based on the dissertation template inspired by the corporate design of the University of Luebeck, which was written by Ronny Bergmann <bergmann@math.uni-luebeck.de> (thesis template, 4. August 2013). The template was adequately modified and extended for the purpose of this dissertation.

Sections 6.4.2 and 6.6 in this dissertation were implemented by a Bachelor's degree candidate, Jan Gröschner, under our close supervision. His Bachelor's thesis [63] included the implementation of first working prototype of the Interaction Editor and the first draft of XML representation of Labanotation (v 0.9.0). The two contributions are based fully on our conceptual design and guidelines for movement profiling.

The passive style of writing is used in most parts of this dissertation. Nonetheless, the first person plural style of writing is also used in some parts to improve coherence and readability. Code snippets appear in the thesis are mainly used to for implementation clarification purposes or explanations.

A number of peer-reviewed conference papers and successful funding proposals related to this research are listed on the next page.

Publications and Successful Funding Proposals

The author's Curriculum Vitae is presented in Appendix A

The author's full list of publications is presented in Appendix B

Some of the ideas, figures, text, and results have appeared previously or are scheduled to appear in the following closely related publications and funding proposals:

- **Altakrouri, B.** and Schrader, A. Dynamic Interaction Plugins Deployment in Ambient Spaces. To appear in 5th International Conference on Human-Centered Software Engineering 2014 (HCSE'2014), September 16 - 18 2014, Paderborn, Germany.

- **Altakrouri, B.** and Schrader, A. Describing movements for motion gestures. In the 1st International Workshop on Engineering Gestures for Multimodal Interfaces (EGMI 2014) at the sixth ACM SIGCHI Symposium on Engineering Interactive Computing Systems (EICS'14), June 17 2014, Rome, Italy.

- **Altakrouri, B.**, Carlson, D., and Schrader, A. Sharing kinetic interactions for mobile devices. In Design, User Experience, and Usability. User Experience in Novel Technological Environments, A. Marcus, Ed., vol. 8014 of Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2013, pp. 327 - 336.

- **Altakrouri, B.**, Gröschner, J., and Schrader, A. Documenting natural interactions. In CHI'13 Extended Abstracts on Human Factors in Computing Systems (New York, NY, USA, 2013), CHI EA'13, ACM, pp. 1173 - 1178.

- **Altakrouri, B.** and Schrader, A. Towards dynamic natural interaction ensembles. In Fourth International Workshop on Physicality (Physicality 2012) co-located with British HCI 2012 conference (Birmingham, UK, September 2012), A. D. Devina Ramduny-Ellis and S. Gill, Eds.

- **Ensembles - "Ambiente MIT Ökosysteme - Adaptive Mensch-Technik-Interaktion im Verbund"**. Project funded by Bundesministerium für Bildung und Forschung (BMBF), IKT2020-Wissenschaftliche Vorprojekte MTI/DW, July 2014 - June 2015.

- **"STAGE: ein anthropometrisch unterstützendes Framework für die physikalische Interaktion mit digitalen Artefakten durch Menschen mit körperlichen Behinderungen - Machbarkeits- und Akzeptanzstudie**. Project funded by Verein zur Förderung der Rehabilitations-Forschung in Hamburg, Mecklenburg-Vorpommern und Schleswig-Holstein e.V., June 2014 - December 2014.

Kurzfassung

Mit jedem Fortschritt im Bereich der Mensch-Computer-Interaktion (Human-Computer Interaction; HCI) werden die natürlichen Interaktions-Technologien (Natural User Interfaces; NUI) bereichert. Trotz zahlreicher neuer Interaktionstechniken, die jedes Jahr vorgeschlagen werden, wird das volle Potenzial der sensorischen und motorischen Systeme des Menschen noch nicht ausgeschöpft. In letzter Zeit sind vermehrt Aufrufe laut geworden, das Potenzial des ganzen menchlischen Körpers für die Interaktionen mit der realen Welt pervasiver und ubiquitärer Umgebungen (ambiente Umgebungen) zu erforschen. Mit zunehmender Anwendung der NUI-Technologien in ambienten Umgebungen ist zu erwarten, dass die Benutzer mit mehreren Techniken gleichzeitig interagieren.

Während NUI-Technologien reichhaltige Interaktionsmöglichkeiten und Alternativen bieten, erzeugen sie auch neue kritische Herausforderungen für interaktive ambiente Umgebungen. Diese Dissertation befasst sich mit drei wesentlichen Herausforderungen: dem massiven Einsatz dynamischer Verteilung von bestehenden und künftigen Interaktionstechniken zur Laufzeit, der langfristigen und adäquaten Aufzeichnung und Verbreitung von Interaktionstechniken und der Anpassung der spezifischen Möglichkeiten der Interaktion in-situ. Diese Herausforderungen werden zusätzlich erschwert durch eine erhöhte Mobilität der Nutzer, die Heterogenität und Verfügbarkeit von Interaktions-Ressourcen und der zunehmenden Diversifizierung körperlicher Fähigkeiten in unterschiedlichen Benutzergruppen (z.B. ältere Nutzer).

Diese Dissertation präsentiert einen neuen Ansatz für die Adaption der Interaktionsmodalitäten einer bestimmte Anwendung zur Laufzeit (in Form von integrierbaren Interaktions-Plugins). Damit werden die Möglichkeiten und das Verhalten eines interaktiven Systems optimal an die körperlichen Fähigkeiten, Bedürfnisse und den Kontext der Nutzer angepasst.

Der Ansatz umfasst ein theoretisches Konzept (*Interaction Ensemble*) mit einer Entkopplung der oft engen Bindung von Geräten, Interaktionstechniken und Anwendungen. Eine Referenz-Implementierung (*STAGE-Framework*) wird zur Evaluation des Konzepts vorgestellt.

Abstract

Advances in Human Computer Interaction techniques continue to enrich Natural User Interface (NUI) research. While dozens of novel NUI interaction techniques are proposed every year, the potential of the human body's sensory and motor systems is not yet fully utilized. Hence, new pressing calls have emerged for exploring the potential of the whole body in motion when interacting with real-world pervasive and ubiquitous computing ecosystems (ambient spaces). Given the adoption of NUI paradigm in ambient spaces, users will be increasingly expected to interact with multiple interactions techniques simultaneously.

Whilst NUIs provide rich interaction possibilities and alternatives, they also introduce critical challenges for interactive ambient spaces. This dissertation aims to tackle three of these challenges; namely, large-scale dynamic runtime deployment of existing and future interaction techniques; long-term and adequate record-keeping and dissemination practices for interaction techniques; and in-situ adaptation of interaction possibilities. These challenges are often fueled by users' increased mobility; the increasingly heterogeneity and availability of interaction resources; and the increasing diversity of the physical abilities of many user populations (e.g., elderly users).

This dissertation presents a novel approach for adapting the interaction modalities available to a given application at runtime (as deployable interaction plugins). Accordingly, the capabilities and behaviour of an interactive system are optimized to fit the users' physical abilities, needs, and context.

The approach includes a theoretical concept (called *Interaction Ensemble*) that relies on decoupling the often tight binding between devices, interaction techniques, and applications. A reference implementation (called the *STAGE* framework) is presented as an evaluation of the concept.

Contents

1 Introduction

1.1. Motivation

Pervasive and ubiquitous computing technologies have paved the way for the development of future interactive environments. Such environments consist of a plethora of interconnected smart objects to realize new context-aware services in a seamlessly integrated physical and virtual world. Pervasive and ubiquitous computing paradigms foster "dissolving traditional boundaries for how, when, and where humans and computers interact" [116]. This enables systems to "weave themselves into the fabric of everyday life until they are indistinguishable from it" [179]. The improved integration of these technologies into commodity devices has set a strong ground for Natural Interactions (NIs) to be preponderant in pervasive and ubiquitous environments such as experience centers and museums [56]. It also imposes new challenges for Human Computer Interaction (henceforth, HCI). Moreover, the HCI research faces challenges from new urging trends such as increasing aging population, turning to more personal computing, increasing mobility, and expanding computing capabilities converging into handheld devices [182]. The goal of HCI research on pervasive and ubiquitous environments is to create user-friendly interaction means for essentially invisible technology. Technology should therefore adapt to the natural interaction abilities and practices of humans.

Human users will continue to interact with pervasive systems using physical body interactions and intermediaries, because major body activities such as touching, holding, and moving physical objects are the foundation of the long evolution of tool use in the human species [71]. Similarly, voice based communication will continue to be used due to its essential role in human communication culture. Those interaction styles form the basis of the definition of NI, which can be shortly devised as voice-based and kinetic-based interactions drawing on Wachs et al. [172]. Therefore, interaction using Natural User Interfaces (henceforth, NUIs) is frequently being proposed solution to support a flow of (inter-)action patterns in the hybrid world similar to the human patterns in the physical space [168].

The definition of NI varies in the literature as noted by Iacolina et al. [79]. Nevertheless, the bulk of existing definitions generally refers to the use of users' natural abilities, practices, and activities to control interactive systems.

Such definitions inherently include activities such as but not limited to gestures, physical and virtual objects manipulations, body movements, and postures [79]. NI resembles closely forms of human's communicative abilities [1] and enables more natural and intuitive communication between people and all kinds of sensor-based devices to enforce interactions that would "feel right".

There is a strong evidence in the literature suggesting that natural interaction style will be an essential part of future of human computer interactions. This interaction style depends on enabling more natural and intuitive communication between people and all kinds of sensor-based devices, to enforce interactions that would feel right, without the encumbrance of a keyboard or mouse [134] [167]. Abowd and Mynatt [1] identified three essential research themes to investigate within the area of ubiquitous and pervasive computing: natural interaction, context awareness, and automated capture and access. Two of them are also reflected in a list of nine challenges for computer science suggested by the UK Computing Research Committee, namely science for global ubiquitous computing (GC2) and memories for life (GC3) [73].

A comprehensive analysis of this investigation is presented in Chapter 5

The interest in natural interaction research has been reflected with tens of studies and novel interaction techniques investigated and developed by HCI researchers each year. Reviewing the Association for Computing Machinery (ACM) published research around this area alone revealed more than 39 new interaction techniques and 53 studies between January 2012 and August 2013. This list expands greatly when other publishing venues and research criteria are considered.

This thesis is motivated by the lack of existing adaptation techniques and strategies adequate for utilizing the human body in its full potential in interactive ambient systems. Currently, interaction adaptation to the user's physical needs and abilities mainly occurs at the level of an individual interaction techniques (micro level adaptation). While micro adaptation is useful, it clearly fails to scale to the vast adoption of NUIs in interactive system and fails to cope with the dramatic shift towards an aging society. Alternatively, this thesis argues for the "full-body-in-motion" as a holistic approach for NI adaptation in interactive ambient systems on a macro level. Macro adaptation fosters simultaneous functioning of multiple interaction techniques in an interactive eco-system to maximize the utility of the physical abilities of users, especially elderly users and users with disabilities, for interactivity purposes.

1.2. Towards NUI in ambient spaces

The HCI research has continued to flourish with an expanding world of interconnected devices and technologies driven by rich interaction capabilities. Centrally, the physical body interactions and intermediaries continue to facilitate users' interactions with real-world pervasive and ubiquitous eco-systems (ambient spaces). NIs foster a set of important interaction qualities [172] including high accessibility, engagement, familiarity, easiness, intuitiveness (clear cognitive association with the functionality performed), come as you are, ubiquity and wearability without requiring long periods of learning and adaptation. Particularly, NI is able, as argued by Wachs et al. [172], to solve a number of challenging aspects in ambient spaces:

- overcoming physical handicaps,
- exploring big data,
- and finally accessing and conveying information, meaning, and intentions while maintaining high sterility, where users are able to embrace such new alternative interfaces and interactions.

The HCI literature is very rich with interaction techniques proposed by the research community. Reported pioneering work on NI pertain to scratch-based interactions [61], accelerometer-based interactions [154], sensor-based interactions [154], and ambient gestures [87]. In the last decade, touch and motion enabled technologies found their way commercially and became widely accessible to the end users. Hence, users in ambient spaces are becoming more acquainted with using different body parts to interact with interactive applications such as gaming (e.g., motion-controlled active play by Microsoft Kinect[1] or the Wii system[2]), data browsing, navigation scenarios (e.g., tilting for scrolling photos as in iOS[3] and Android[4] devices). The advancement and increasing number of available and newly developed interaction techniques impose new challenges on large scale dynamic and flexible deployment of those techniques [137].

Despite this innovation, Buxton's claim holds partially true [25]. He argued that interactions are designed for well-developed eyes, long right arms, uniform-length fingers, and ears. Nonetheless, it lacks legs, a sense of smell or touch. Moreover, he claimed that human operated machinery and computer systems make poor use of the human's sensory and motor systems potential. To this day, some interaction techniques have gained much more attention

[1] http://www.microsoft.com/en-us/kinectforwindows/, latest access on 01.04.2014.
[2] http://www.nintendo.com/wii, latest access on 01.04.2014.
[3] http://www.apple.com/ios/, latest access on 01.04.2014.
[4] http://www.android.com/, latest access on 01.04.2014.

and focus by the community, e.g., multitouch interactions have gained much more attention than others (e.g., feet interactions) [42][56]. Furthermore, the development of interactions is still very limited in terms of covering the increasing diversity of user population physical abilities, needs, and context.

Recently, a few new studies have opened alternative directions to explore new bodily potential in designing for the body in motion, notably to motivate the user population to take active parts in interactions and influence motor development as in Kinesthetic Interaction by [56]. Opening new alternative explorations for interaction in ambient spaces is very important to maximize users' interaction abilities in a given context. For instance, interactions based on hand gestures are well suited for rather precise input. Wiping gestures are good for panning a map for larger distance, but may lead to ergonomic problems such as arm fatigue. Foot interactions, in contrast, provide an intuitive and less exhausting modality for continuous input. For example, feet interactions borrowing from the striking metaphor (i.e., pushing the body weight over the respective foot) are intuitively used to navigate in real space [42].

Despite novel research contributions to this area in the last two decades, the fusion and composition of NI techniques are still rather unexplored areas. The combination of hand and foot input for example has gained only little attention according to Daiber et al. [42]. Principally, distributed and modular device interfaces, enabled by accumulative hardware ensembles, proved useful for interactions where additional interface elements can be added as needed to devices, like a larger screen, a mouse, speakers, a keyboard, a projector, and an e-ink screen.

The concept "Interaction Ensembles" is defined in Section 3.4

Although ambient spaces require highly adaptive and modular interaction approaches, much of the literature focuses on using a limited part of the body. Interactive systems that incorporate the gross motor skills and utilize the kinesthetic sense have not been thoroughly investigated despite the growing number of implementation examples [56]. Lately, a strong emerging motivation to explore new potential in designing for the whole body in motion appears, as in Kinesthetic Interactions [56]. Against this background, users are expected to interact with multiple interaction techniques simultaneously by employing multiple body parts and different motor skills. Hence, this thesis argues that *NI is expected to play not only individually but also as part of an ensemble (i.e., a united collection of interaction techniques).*

Interaction with ambient spaces is becoming increasingly challenging as user population grows to include users with varying intrinsic sensorimotor capabilities, ranging from injuries, aging, or other disorders. Interest in specially tailored applications for health related sensorimotor deficits have come to the fore. Lately, home care research and industry are opting for more intuitive support for elderly and disabled people, for instance elderly people with physical limitations are actively using Wii for fun and rehabilitations

[20]. Nevertheless, this effort is still considered modest as vast research studies, e.g., surface computing studies, are still made for the general audiences with little focus on older adults [134].

A review of the literature reveals an extensive effort in the area of user interfaces adaptation. Adaptation aims at changing and modifying user interfaces to adequately correspond to the context of use. The changes may include the interface presentation (including media and interaction techniques, layout, and graphical attributes), dynamic behaviour (including navigation structure, dynamic activation, and deactivation of interaction techniques), and content (including texts, labels, and images) [132]. The context of use refers usually to user-related context (such as preferences, goals, physical state, and emotional state), environmental context (such as location, lighting conditions, and available devices), and social context (such as collaboration and privacy) [132]. One of the well-established concepts is plasticity [26]. This concept refers to the capacity of an interactive system to tolerate changes in the context of use while retaining usability based on adapting the graphical user interface according to three factors (input, output, and platform). The WWHT framework [143], on the other hand, is based on a rule-based system, which matches different communication channels to a given context model based on four levels of adaptation (What, Which, How, Then).

In computer science, the term context refers to "any information that can be used to characterize the situation of an entity." [45]

Despite being a hot HCI topic, most available adaptation approaches fail to satisfy four enduring challenges drawn from the natural characteristics of ambient environments, presented by Pruvost et al. [137]:

- Heterogeneity and Distributivity: The interaction eco-system contains a variety of interaction devices with various capabilities.
- Dynamic Media Mobility: Interaction capabilities are highly dynamic as interaction devices may join or leave the ambient space at anytime.
- User Mobility: The user mobility in ambient spaces challenges the interactive system attention to the user's interaction needs.

Moreover, most adaptation approaches focus on interface issues such as information presentation but not the interaction techniques per se. There is currently a growing interest in investigating interaction adaptation. Recent work by Pruvost et al. [137] focuses on interaction adaptations in ambient environments. They have suggested the concept of interaction ontology where semantic information about the interface, user and the context are used for interaction adaptation. They focus mainly on the structural adaptation of user interfaces and the adaptation of running interaction dialogs.

On the other hand, the growing interest in NI techniques is also reflected in a number of studies on their social acceptability. Rico and Brewster [141] investigated the social acceptability of gesture-based interactions with mobile phones in private and public settings. Their results provide researchers with concrete tools to assess the social acceptability of multimodal interaction techniques at an early stage of development. Moreover, research on end-user elicitation of motion gestures was reported by [146]. The authors provided a strong evidence for the importance and acceptability of kinetic-based NI, as a high percentage of respondents (up to 82%) were willing to use motion-based gestures at least occasionally. Triggering adequate actions to the user's natural input, which aims to insure the acceptability of NIs, is not sufficient. Additionally, interactive systems in ambient spaces should be able to infer the correct interpretation of actions in context. Moreover, they should be able to convey to users information that is relevant to them, appropriate to the situation, adequately articulated in terms of simplicity and complexity, and provide such information timely [160].

One of the most demanding user populations for NI is the senior citizen population, due to the notable effect of aging in one's physical and motor abilities. Elderly adults experience an overall slowing of movement and major problems with fine motor activity and coordination [111] [134] often resulting in inaccessible conventional interfaces (e.g., mobile interfaces) according to Kane et al. [86]. Paradoxically, whilst NIs provide rich interaction possibilities and alternatives, they are also affected by physical impairments [111] including visual, hearing, and mobility. Impairments such as arthritis, paralysis, and Parkinson's disease contribute to vast range of symptoms affecting kinetic interactions greatly such as limited range of motion, pain, tremors, impaired balance, and gait. While assistive living technologies such as hypermedia interfaces, increased intelligence of home appliances, and collaborative environments are now converging and representing an important enabling factor for elderly support community environments [20], the absence of the right interaction strategies and possibilities may lead to cause great burden on the elderly user.

1.3. Thesis focus

This research focuses on on Kinetic-based NI, therefore other types of NUI techniques are excluded intentionally from the discussion throughout this dissertation. As Natural Interactions between the physical and virtual spaces widely take place by means of gestures, body movements, and physical and virtual objects (i.e., artifacts) manipulation, it is important to acknowledge that the core concepts covered in this dissertation can still be applied to relevant and closely related interaction types in ambient spaces such as tangible interactions, interaction with 3D interfaces, etc.

1.4. Thesis statement

Despite rapid innovation in NI techniques, it is well understood that user interface adaptation in ambient spaces remains a challenging problem [137], due to issues related to heterogeneity and distributivity; dynamic media mobility; and, user mobility. This dissertation contends that kinetic interactions are currently hindered due to the lack of:

1. Systematic consideration of the increasing diversity of user populations in ambient spaces. In particular, existing approaches fail to address users with varying intrinsic sensorimotor capabilities and fail to comply with the general trend of designing for the whole body in motion; and,

2. Effective means for documenting, adapting and deploying interaction techniques. Interaction techniques are currently hard-wired into applications, leaving few possibilities for the integration of new NI techniques at runtime. Consequently, the task of sharing interactions becomes unrealistic in many scenarios.

This dissertation argues that *it is vital that anthropometric based analysis of NIs leads to match users' physical abilities and disabilities to the current environment and interaction context.* Interactive eco-systems should enforce better performance and integration of users within their known physical abilities, which can also be increasingly useful in physical therapy and rehabilitation, e.g., maintaining and improving mobility, flexibility, strength, gait speed, and quality of life. Fogtmann et al. [56] call for conceptual frameworks to identify unexplored possibilities when designing interactive systems addressing the body in motion. Hence, we argue for an anthropometric approach for describing, discovering and presenting interaction techniques. Moreover, the *theoretical gist of this research is to study anthropometric driven ensembles of natural interaction techniques. This novel approach fosters de-coupling the close binding between devices, interaction techniques, and applications. Hence, the approach aims at designing interactions "for all" instead of focusing on a limited population percentile.*

1.5. Research issues

We therefore endeavor to investigate the theoretical concept of NI Ensembles extensively, which we believe will be part of the enabling technology for ambient spaces.

The definition of the Interaction Ensemble concept is presented in section 3.4

This novel area of research is challenged by several pressing issues, only some are pressing and can be discussed in the scope of this dissertation:

- The literature's lack of published research on motion-based interaction primitives classification and design space of motion-based interactions, despite some recent research effort to end-user elicitation of motion gestures as reported by [146] (discussed in Chapter 2);
- The lack of standardized decimation and sharing processes of NI techniques that hinder real world deployment (mainly covered in Chapter 3 and Chapter 4);
- The lack of macro interaction adaption strategies in interactive eco-systems (mainly covered in Chapter 3); and,
- The absence of standardized and formalized documentation languages and documentation strategies for NI (mainly covered in Chapter 5 and Chapter 6).

We have identified three closely related research issues to be solved for better adoption of natural interactions in ambient systems:

1. Assessment of anthropometric physical abilities and disabilities;
2. Interaction ensembles and orchestration; and,
3. Community-based designing and sharing of interaction techniques.

1.6. Dissertation structure

The definitions of AAL is introduced in section 2.1.3

The dissertation is structured in seven chapters. Chapter 2 substantiates the dissertation's motivation and approach by providing an overview over various topics related to NUI in the context of the dissertation's focal points. Section 2.1 begins by discussing related work in Ambient Assisted Living (AAL) and covering, in addition to a generic overview, mainly three topics including: the aging impact on AAL (2.1.2), AAL application areas (2.1.3), and AAL social implications (2.1.4). Next, section 2.2 covers related work in NUIs. It starts by discussing various definitions of NUI in detail (2.2.1). Then it discusses the natural aspects of NUI from different viewpoints (2.2.2), the movement towards Whole Body Interactions (2.2.3), gestures for interactions (2.2.3), movements and interaction design (2.2.3), the social acceptability of NUI (2.2.4), the disability challenges for interactions (2.2.5), and the adaptation challenges and issues in NUIs (2.2.6). Finally, the chapter is concluded by an outlook section (2.3).

Based on the background and related work presented in Chapter 2, the core approach towards an anthropometric framework for NI Ensembles is presented in Chapter 3. The chapter begins with section 3.1, which covers a brief overview over the traditional "one-design-fits-all" design approach and the shortcoming of this approach. Next, section 3.2 argues for de-coupling interactions in ambient spaces. Section 3.3 introduces relevant concepts to the theoretical concept of this thesis including input tasks and interaction primitives. Shortly after, section 3.4 presents the core definitions of interaction ensemble and interaction plugin. To further elaborate on those definitions, a full walkthrough user scenario is presented in section 3.5 and six different interaction ensemble cases are presented in section 3.6. Building and orchestrating Interaction Ensembles as a theoretical concept is further presented in section 3.7. The section is split into four subsections discussing; first, building Interaction Plugins (3.7.1); second, anthropometric driven matching and presentation of Interaction Plugins (3.7.2); third, on demand wiring of interaction resources (3.7.3); fourth, community-based designing and sharing of IPs (3.7.4). Finally, the chapter is conceded by an outlook section (3.8).

Building on the approach and conceptual design presented in Chapter 3, Chapter 4 discusses a NUI deployment in ambient spaces. Section 4.1.1 discusses the dynamic component integration, context modeling, and deployment with the Dynamix framework (4.1.2). The architecture and implementation of the STAGE system, as a realization of the conceptual design, are presented in Section 4.2. In section 4.5, the implementation of STAGE-enabled interactive applications is presented, including: the interaction manager (4.5.1), ability manager (4.5.2), interaction profile manifest (4.5.3), and the interaction sequence of STAGE-enabled applications (4.5.4). Furthermore, the chapter presents the implementation of Interaction Plugins in section 4.3. Next, the provisioning of IP is presented in section 4.4. In addition to the implementation details, a performance evaluation of IP based on the Dynamix framework is presented in section 4.6. Moreover, a number of IP examples and showcases are presented in section 4.7 including the 3Gear hand motion IPs (4.7.1) and the AmbientRoom application (4.7.2). Finally, the conclusions of this chapter are presented in section 4.8.

A detailed review on documentation and learning practices for interaction in NUIs is presented in Chapter 5. The chapter starts with an introduction in section 5.1. Section 5.2 presents a tailored survey targeted at designers and users. Next, section 5.3 presents an analysis of a large sample of recently published multitouch- and motion-based interaction papers. Section 5.4 reviews a number of multitouch- and motion-based application market places. The three aforementioned sections present different results and observations of the investigation areas. The chapter also features a discussion section (5.5) where various topics are discussed, such as the designers' documentation habits (5.5.1), documentation styles (5.5.2), the importance of documentation (5.5.3), and various challenges of documentation and learning in ambient spaces (5.5.4). Finally, the emerged conclusions of this chapter are presented in section 5.6.

The issue of NUI documentation as an essential cornerstone towards the vision of this dissertation is discussed in Chapter 6. The chapter starts with a literature review on this field covering two topics in particular, namely movement documentation in practice (6.1.2) and movement documentation for interactions (6.1.3). Documentation-related tools and languages are discussed in section 6.2. Next, section 6.3 discusses in detail the thesis approach towards documenting NI techniques. Accordingly, section 6.4 presents the design and implementation of movement profiles including a discussion about Labanotation (6.4.1), a tailored XML representation of the movement profile based on Labanotation (6.4.2), and a set of examples and showcases (6.4.3). Next, physical ability profiles are discussed in section 6.5. In addition, a tool called Interaction Editor for authoring NI is presented in section 6.6. Finally, the chapter is concluded in section 6.7.

Finally, Chapter 7 presets the general conclusions of this dissertation. The contributions and achievements of this work are presented in section 7.1. A detailed discussion of possible future work and possible improvements are presented in section 7.2.

2
On Natural User Interfaces

This chapter presents an overview material and related work regarding NUI paradigm as the main topic of the thesis. While the study of this paradigm can be tackled from many different viewpoints, this chapter only discusses the most relevant views and research themes strongly related to the core context of this dissertation. The chapter is split into two parts. The first part, presented by section 2.1, starts by discussing related work in Ambient Assisted Living (henceforth, AAL). It covers a general overview on Ambient Intelligence (henceforth, AmI) and AAL including various definitions, ambient space categories, application areas and scenarios, etc. For more details, the section is split into subsections discussing aging impact on AAL (discussed in 2.1.2), AAL application areas (2.1.3), and finally closing with social implications of AAL (2.1.4).

The second part, presented by section 2.2, offers a detailed discussion about seven closely related NUIs topics to the motivation of this dissertation. First, the section begins by discussing various definitions on this interaction paradigm (2.2.1). Second, a discussion about various viewpoints and criticisms is presented (2.2.2). Third, related work towards whole-body interactions is then presented (2.2.3). Fourth, gestures as a research topic and gestures categorizations are discussed (2.2.3). Fifth, a dedicated discussion about movements for interaction design is presented (2.2.3). Sixth, the social acceptability of NUI is discussed (2.2.4). Seventh, the physical abilities heterogeneity challenges to NUIs are discussed (2.2.5). Eighth, the adaptation challenges and issues in NUIs are presented (2.2.6). Finally, the chapter presents an outlook section (2.3).

2.1. Related work in Ambient Intelligence

2.1.1. Ambient spaces

In the late 80s, Mark Weiser coined the term *"Ubiquitous Computing"* and put the first cornerstone to establish what he thought to be the future of computing. Weiser defined ubiquitous computing as "the method of enhancing computer use by making many computers available throughout the physical environment, but making them effectively invisible to the user"

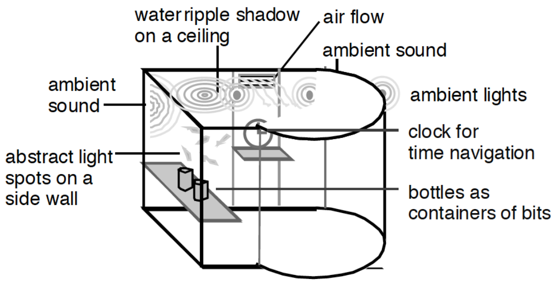
[178]. For the last two decades, a lot of research and development has been done in this area and Weiser's vision is becoming rapidly a reality [131]. Soon later, a new related term emerged, namely *"Pervasive Computing"*. Often both terms are used interchangeably [148], but at the same time a lot of researchers form both sides praise the conceptual differences between the two [109]. The Pervasive Computing research focuses on building computing modules based on the context of the environment and aims at providing invisible computing services available to users. The Ubiquitous Computing research, on the other hand, relies on the mobility of computing services and the reduced sizes of computing devices. Hence, it provides pervasive environments to a human user on the go (i.e., supports user mobility). The conceptual differences between both fields do not affect the discussion of this thesis, and thus both are used interchangeably.

Real-world pervasive and ubiquitous computing technologies are steadily proliferating into everyday objects, devices, services, and environments. Nowadays users are offered unprecedented computational capabilities and continuous interconnectivity. Thus, users are becoming aquatinted with in-situ and context dependent services in personal, collaborative, and public spaces.

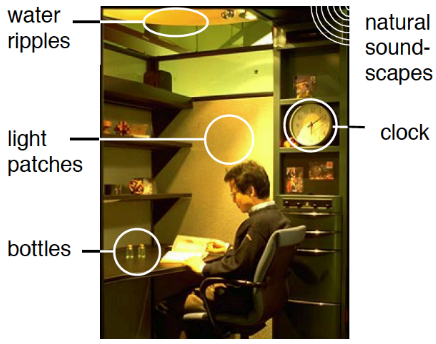
The term AmI was first coined by the European Commission in 2001, based mainly on the emergence of ubiquitous computing, sensor network technology, and artificial intelligence. Soon later, it was widely and internationally adopted by researchers [147].

The main vision of AmI promises to create ambient spaces, which are unobtrusive, interconnected, adaptable, dynamic, embedded, and intelligent. AmI facilities a direct communication with surrounding devices, clothes, furniture, and everyday objects. AmI is currently used and applied in wide range of application and usage scenarios including ambient homes, assisted living and domestic care, health care, shopping and business, museums, and tourism. An extensive review of AmI use and its application areas was presented by Sadri [147]. Probably one of the earliest examples of ambient personal (i.e., private) spaces is the ambientRoom project [80] shown in Figure 2.1. This project demonstrated the strength and potential when integrating ambient media with architecture spaces and objects. Figure 2.1 illustrates a variety of ambient actuators, e.g., sound and light, and controls that mediate a flow of diverse media content.

There have been a number of attempts to apply AmI in the domestic context in different directions following the ambientRoom project, such as the Microsoft's Home of the Future project [135], the Adaptive House [38], Synapse [157], GENIO [60], and many more [131]. For instance, the GENIO project aimed at household appliances such as fridges, washing machines, ovens, sensors, and heating devices to be connected and controlled by a central unit and direct communication with the user using natural language



(a) Schematic diagram

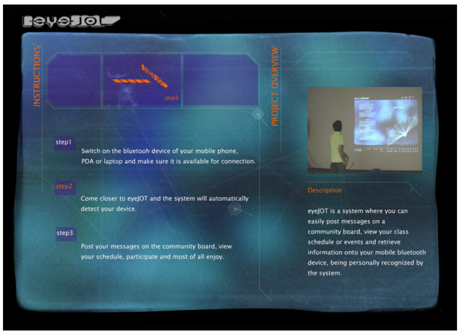


(b) Ambient media displays and controls

Figure 2.1.: The ambientROOM project as example for integrating ambient media with architectural spaces (from [80], used with permissions from the author)

[60]. The Microsoft's Home of the Future project in Redmond, Washington is a rich example to illustrate the use of different ambient technologies and interfaces [147]. It is a living example of future ambient homes where every inhabited space, such as living rooms, bathrooms, bed rooms, kitchens and its appliances, etc., responses to the user context, presents information, and allows interaction. Technologies such as interactive surfaces, embedded displays, RFID tags, and embedded sensors are weaved together as basic building blocks for this project. In the "Microsoft's Home of The Future" project [135], two of many demonstrating scenarios are the Microsoft Home's kitchen and the teenager room. The former demonstrates an intelligent kitchen counter, where on recipes are suggested and projected. The latter demonstrates a teenager room that adapts to the teenager's mood and changes its decoration accordingly. The room also reveals social updates on the room's wall, in order to keep the teenager up-to-date and connected to friends.

Ambient personal spaces range from mobile spaces as in handheld devices to more situated or fixed spaces as in ambient homes. The Daidalos project [162] aimed to investigate ambient spaces for mobile users. In our previous work on the eyeJOT project [4], we presented a new context-aware smart campus information system. eyeJOT combined ambient wall-sized displays with location-aware and context-sensitive information sharing on mobile devices. It reacted dynamically to the user proximity and awareness of surrounding crowd to customize the information presentation accordingly, in order to convert a public wall into a private space for a particular user. In fact eyeJOT allowed to seamlessly alter between different private and public settings based on the measured user's proximity with the wall. By getting closer to the wall, i.e., entering the 150 cm intimate proximity distance, eyeJot activates different levels of information abstractions and reveals different interfaces to the user to interact with.



(a) Instructional welcome interface (Public Space)

(b) Community messages interface (Public Space)

(c) Personal calendar (Private Space)

Figure 2.2.: eyeJOT - Ubiquitous context-aware campus information system interfaces (from [4])

On the other hand, public ambient spaces have been the subject of many research projects. Vogel and Balakrishnan [170] developed an interaction framework for sharable and interactive public ambient displays. The PRISMATICA project [169] adopted a computer-vision approach to detect situations of interest in busy conditions, especially in public transport environments. The Mobilife project [180] focused mainly on the user's shared cognition in ambient spaces. Furthermore, two of the most popular investigated scenarios in public ambient spaces is to enhance the shopping experience through highly responsive shops and ambient advertisement [40] and smart exhibition spaces for museums and galleries as in our previous work on the the Medient system [9] and the Mobile HolstenTour project [6].

Another category of ambient spaces is collaborative ambient spaces, which offer collective services for multiple users in an ambient environment simultaneously. AmIART project [43] was presented as a prototype for collaborative learning and experimental environments. Moreover, the LORNAV project [49] was reported to target different multimedia objects visualization and aggregation in a 3D virtual environment.

The expanding spectrum of AmI projects triggered multiple research directions to investigate the social acceptability of AmI techniques as in [47]. The social acceptability of AmI techniques is conditional to the ability to facilitate human contact, inspire trust and confidence, and to be controllable by ordinary people [47]. More recently, AmI systems were characterized as being invisible (i.e., embedded in every days object), mobile (i.e., carried around), spontaneous (i.e., spontaneous communication), context-aware (i.e., awareness of local environments), anticipatory (i.e., acting on their own behalf without explicit extrinsic requests), interactive (through natural interfaces with users), and finally adaptive (i.e., adequate situational reactions) [177] [18]. Due to those characteristics, the impact and challenges of AmI systems on social acceptability are still undergoing research themes.

Of a particular interest to this dissertation are the challenges of user interfaces in AmI. Ambient interfaces use the whole environment to facilitate the interactions between AmI systems and users [65]. In addition to the previously mentioned characteristics of AmI systems, information presentation in ambient spaces is subject to subtle changes of the physical context such as light changes, sounds variations, or movements. Hence, ambient interfaces deviate from the classical graphical interface paradigm (where interfaces reside on a computer screen) to the natural interface paradigm (where interfaces resides in physical objects and surroundings), in oder to facilitate a more intuitive and easy to use interactive systems [64].

In his work, Gross [65] presented general guidelines for designing ambient interfaces. The guidelines are collective efforts from different research resources and include a number of very interesting properties that an ambient interface should have. According to the guidelines, an ambient interface is characterized as being effective, efficient, safe, easy to learn, easy to remember, visible, adequate, and participatory. Additionally, an ambient interface should have a good utility and should provide an adequate feedback, constrains, mapping, and a consistent functionality. In the context of this work, the aforementioned guidelines can be used as evaluation criteria for assisting and evaluating Ensemble-enabled interactive systems.

Gross's full guidelines are briefly described in Appendix C (Table C.1)

2.1.2. Aging impact on AAL

In this section, the impact of aging on the development of AAL systems is examined. The demographic change towards aging societies is clearly noticeable with unprecedented population of elderly. The most recent statistics on the European Union (EU) population structure and aging are reported by the European Commission Eurostat [52]. The statistics reveal that this demographic change is clearly fueled by a combination of low birth rates and a higher life expectancy. This is transforming the EU population structure with a significantly greater share of older persons, especially in the coming decades. For example in the EU-27 [1], Germany has recorded the highest population percentage of elderly (older than 65 years old) with 20.6% and the lowest percentage of young people (0 to 14 years old) with only 11.5%. It is obvious that this demographic shift did not begin recently, but it began several decades ago. The population pyramid for the year 2011 shows a 3.7% decrease of young population share and 3.6% share increase for elderly population for the last two decades. Moreover, the median age of the EU population has increase from 35.4 years to 41.2 years in the period between 1991 and 2011.

The latest population projections for the EU-27 member states reveal that in 2060 all states will have a much older age structure than now with a slight increase in the population. The median age is projected to raise to 47.6 years (41.2 years in 2011). The member states will be facing a steady decline of the working population and an increase of old population, with a population share of 29.5% compared to 17.5% in 2011. Moreover, the projections show that the age 80 years and over will triple by 2060, hence the health care dependency percentage is projected to double (from 26.2% in 2011 to 52.6% in 2060). Figure 2.3 shows the population structure projections for the EU-27 member states for young and old people.

More general predictions on global demographic changes were anticipated by the Organization for Economic Co-operation and Development (OECD). The OECD predicts that, by the end of this decade, old population living in institutions will have increased by 74% (Japan), 61% (Canada), 33% (the US), 26% (Germany), 29% (France), 27% (Sweden), and 18% (the UK) [159]. The fastest growing group of population in the US is the age group over 65 years. By the year 2020, it is expected that the population share of this group will present 1 in 6 of the total population [39]. It is also predicted that fast growth of people with disability, in the same year, will be recorded with expected growth by 74% in Japan, 62% in Canada, 54% in France, 41% in the

[1] Belgium (BE), Denmark (DK), France (FR), Germany (DE), Greece (EL), Ireland (IE), Italy (IT), Luxembourg (LU), Netherlands (NL), Portugal (PT), Spain (ES), United Kingdom (UK), Austria (AT), Finland (FI), Sweden (SE), Cyprus (CY), Czech Republic (CZ), Estonia (EE), Hungary (HU), Latvia (LV), Lithuania (LT), Malta (MT), Poland (PL), Slovakia (SK), Slovenia (SI), Bulgaria (BG) and Romania (RO)

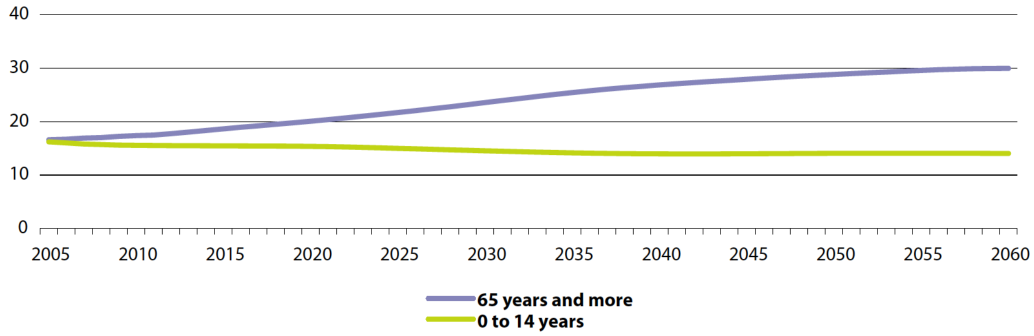


Figure 2.3.: Proportion of the population aged 0-14 and 65 years and more, EU-27 (% of total population)(from [83])

US, and 29% in Sweden [159]. On a global scale as well, the World Health Organization (WHO) projections suggest a 2 billion global estimate of elderly people in 2050 that is approximately 3-fold increase as in the year 2000 [127].

This demographic shift clearly puts an increasing demand on health and social care providers [127][97]. Hence, the market potential of AAL solutions and ICT for aging well is clearly visible, especially for the aging population. For instance, europeans over 65 form a huge market potential, with estimate wealth and revenues of over EUR 3,000 billion [159]. It is estimated that AAL related home devices and smart home appliances users will triple by 2020 with up to 37 million user [159].

2.1.3. AAL application areas and challenges

AmI strongly correlates with enhancements of existing technologies and research fields, including ubiquitous computing, intelligent user-friendly interfaces, and artificial intelligence [147]. One of the main application areas in AmI is AAL. The definition of AAL is formulated around applying AmI technologies to enable people with specific physical requirements, e.g. handicapped or elderly, to longly sustain independent living in their preferred environment [97]. AAL solutions are often adopted for emergency assistance, task execution monitoring, mobility assistance, medication assistance and reminders, etc. In fact, the EU strategy [159] calls for adequate solutions to address daily and independent living such as: social communication; daily shopping, travel, social life, public services; safety; reminders; user-friendly interfaces; telecare and telemedicine; personal health systems; support for people with cognitive problems; and, support for more efficient workflows in care.

According to Kleinberger et al. [97] assisted living solutions should comply with three major requirements in order to facilitate and boost enormous potential for people suffering from all kinds of disabilities. Assisted living solutions and services should be ambient and unobtrusive to reach a high user acceptance, adaptive to personal and environmental context to fulfill individual needs, and accessible to enhance usability. The literature is rich with many projects in the area of AAL as shown in [147]. Figure 2.4 illustrates the wide range of AAL services in home scenarios including emergency assistance, autonomy enhancement, and comfort. In its 7th framework, the EU commission funded AAL projects in different application domains including: mobility, fall detection and prevention, cognitive support, daily-life activities support, service and social robotics, and open platform and tools for AAL [53]. In particular, two prominent funded EU projects are the UniversAAL[2] and AALIANCE2 (a continuation to the AALIANCE project)[3]. The UniversAAL project aimed at the AAL research and development areas, in particular it aimed at integrating the outcomes and features of many AAL projects. The AALIANCE2 project aims at AAL solutions based on advanced ICT technologies for aging and wellbeing of europe's older generations.

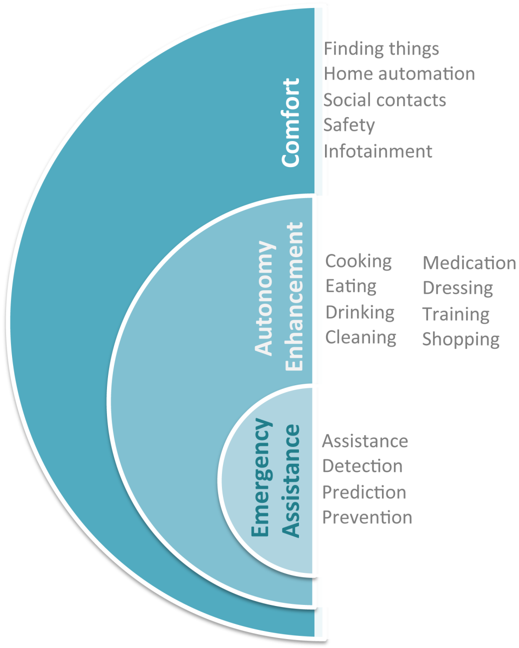


Figure 2.4.: Ambient assisted living in home care system domain (adopted from [18])

[2] http://www.universaal.org/, accessed on 29.04.2014
[3] http://www.aaliance2.eu/node/2, accessed on 29.04.2014

Funded by the German Federal Ministry of Education and Research (BMBF), the SmartAssist platform [153] aimed at developing and realizing an AAL platform for autonomous living, with a direct link to the social network available to the inhabitant both in home and on the move. The platform offers an online service portal; an in-home sensor network; a data collection server; and, an extensible mobile infrastructure called Ambient Dynamix[4]. Wide range of home activities, such as temperature, humidity, movement, location, water and electricity consumption, can be detected and collected at the server side, in order to infer knowledge about the user's health status. Moreover, different mobile and stationary services can also be triggered by the collected context.

One of the main challenges for implementing innovative AAL systems is the lack of adequate business model to govern and strengthen the cooperation between the different AAL stakeholders such as developers, service providers, medical device manufacturers and the housing industry. Hence, AAL solutions and platforms should ultimately ensure modular expansion, self-integration into the environment, and customization to the individual's needs and context [53].

Building AAL platforms and systems is challenged by a following of pressing issues according to Kung and Furfari [99]. The integration of innovation in platforms, where research results are integrated into a platform to maintain and coordinate the AAL development activities; the integration of transversal features including scalability, quality of service, liability, and confidentiality; interoperability of interfaces cause by the diversity of AAL domains; technology independence, where switching between technologies and platform should be possible without changing the application itself); and the support of multiple business models, which if not handled correctly will ultimately result into varying AAL requirements and specifications.

2.1.4. AAL social implications

In this section, we cover the social implications and acceptability of AAL in general and the same implications of user interfaces of the acceptability. We further extend this discussion in section 2.2.4 by primarily discussing on the acceptability of NUIs.

The privacy and ethical implications of AmI systems continue to be the subject of various investigations. The ubiquitous, pervasive, and invisible nature of wide range of AmI technologies triggers many concerns and questions, many of which are still open. In their recent work, Wright et al. [186] considered different scenarios on security and privacy threats of ambient environments and AmI technologies. Four threats were named as the dark

[4]http://ambientdynamix.org, accessed on 24.03.2014

scenarios, which have been presented and discussed in details. Health, mobility, identity and other issues have been covered in those scenarios.

Kemppainen et al. called the community for guidelines regarding designing and implementing AmI systems and technologies for people with disabilities [92]. Rouvroy [144] called for interdisciplinary investigations into AmI issues from the law point of view. Rouvroy's paper investigated the EU frameworks for privacy and data protection and questioned their applicability in AmI systems. From one hand, the EU regulations grant individuals the right to own and control personal data and protection of identity. On the other hand, AmI technologies and systems hold their invisible and pervasive nature while seamlessly sense the environment and collect context information, in order to provide adequate services to individuals. As such, technology weaves into everyday object and space. Hence, the collection and accessibility of information in AmI becomes very complex to be controlled and managed by individuals.

The social acceptability of AmI systems varies according to the targeted context and application area. As part of the e-Sense project, one of the biggest EU-funded projects in the 6th framework on capturing ambient intelligence through wireless sensor networks, a study about AmI acceptability was conducted [147] in three scenarios: personal; health; and industrial. The study shows that continuous monitoring of users and their activities for personalized services was perceived negatively and is thought to be too intrusive. Home health monitoring for better diagnosis and communication between patients and health provers was perceived positively generally, but still with concerns about responsibilities and liability. Finally, tracking and monitoring goods was perceived positively due to impact on increasing productivity.

A deep look at the social implications of AmI spaces was conducted by Bohn et al. [23]. They have identified three main characteristics for any AmI system to comply with: (a) Reliability; (b) Accessibility, and (c) Transparency. Hence, a number of challenges emerge from those characteristics:

- Readability challenge: an AmI system should maintain maintainability, predicability, and controllability;

- Delegation of Control: delegating content and system controls are required in ambient space, but at the same time adequate accounting mechanisms should be provided (who is responsible if something goes wrong?);

- Social Compatibility: ambient spaces and technologies should satisfy and meet the social needs of the society such as transparency (keeping track of micro actions, decisions, and their implications on the user), knowledge sustainability (maintaining useable knowledge in a very dynamic environment), fairness, and universal access; and,

- Acceptance challenge: The feasibility (i.e., prevailing self-confident and technophile attitude among researchers in the field) and credibility (e.g., claiming to simplify our lives) of AmI and ubiquitous computing are often critically challenged. Those criticisms certainly induce a credibility gap. Moreover, the impact on health and environment is a big challenge, especially with no predictions on energy and material consumptions, disposal, etc. More issues on the understanding of the relationship to the world and its everyday object play an important role in the acceptance of AmI as well.

User interfaces are one of the key factors for successful AAL solutions. Interfaces for AAL face a number of challenging problems: (1) Adaptivity problem where the systems have to adapt to the context at runtime; (2) User interface accessibility problems, especially magnified for users with special physical needs and requirements; and, (3) Heterogeneity of devices and services, where integration of services and different technology interfaces should be managed seamlessly [97]. These challenges are clearly visible in many reports about the mismatch between user needs and accessibility and provided solutions in the Information and Communications Technology (ICT) market. Recently, it was reported that ICT equipments and services are not currently adequate to European users (according to 60% of surveyed participants over 50 years old) [159]. Natural user interfaces, covered extensively in this section, may offer great help to tackle the accessibility problem, where multimodal interfaces and full body movements are utilized to create engaging and accessible usage experiences [97].

2.2. Natural User Interfaces in AAL

Physical body interactions and intermediaries are primary facilitators for users' interactions with ambient spaces, not necessarily because the technology requires to do so, but because major body activities are the foundation of the long evolution of tool use in the human species [181]. Currently, natural interfaces are one of the most frequent solutions being proposed to support the flow of (inter-)action patterns in these hybrid environments. NI techniques enable human users to interact with the physical space using familiar physical body interactions and intermediaries [172]. This is clearly visible through the broad adoption of NI techniques as a primary source of interaction within ambient spaces, leading to a rapid increase in the number of techniques proposed by research efforts. On an industrial level, this interaction paradigm is considered a new excitement for many industrial firms [126]. This is particularly boosted by advance developments in processing power, reduced computing costs, increasing memory capacity, increasing quality of vision systems and techniques, and improved sensing devices. The advances facilitate the possibilities to create new devices and systems controlled by gestures and body movements, and go further beyond. Gestures describe situations where body movements are used as a means to communicate to either a machine or a human (revised from Mulder definition of hand gestures [120]). Gestures are essential part of human communication, which comes in different forms such as speech, gestures, facial expressions, and bodily expressions [120]. Gestures undergo different phases. In his work, Quek [138] defined the gesture lifecycle by by three to five phases: preparation, prestroke hold, stroke, poststroke hold, and retraction. Both preparation and retraction refer to movements to bring and return the body part to position and rest respectively. If existed, both prestroke and poststroke serve as timing functions.

Buxton claimed that computers are hardly making any use of the potential of our human sensory and motor systems (introduced in section 1.2). About two decades later, his claims remain valid to a large extend, especially the extremely poor use of the potential of the humans sensory and motor systems as in human operated machinery (e.g., automobile) as noted in Fogtmann et al. [56]. This limited interaction landscape has encouraged researchers and research projects to drive more innovations in the interaction paradigm. Pioneering work on NUI have been reported in HCI literature, ranging from scratch-based interactions [61], accelerometer-based interactions [154], sensor-based interactions [154], ambient gestures [87], etc. However, the interaction paradigms have not gained an equal attention by the community. For example, multitouch interactions have gained much more attention and advances then feet based interactions [42].

Against this background, users in ambient spaces are becoming principally more acquainted with using limited set of body parts (especially hands, arms, and fingers) for interactions with interactive applications, e.g., tilting for scrolling photos. The use of a larger set of body parts for interactions is intimately connected with the expansion of HCI research towards the natural interaction paradigm, through a continuous development of new and novel interaction techniques, as in touch-based and motion-based technology.

2.2.1. Natural User Interface definitions

NUIs capture the user's interactions with physical devices, which are then translated into digital commands, as defined by both Gross and Wisneski et al. [65]. A wide range of physical devices can be considered within the scope of NUI including but not limited to switches, buttons, wheels, touch-enabled surfaces, and cameras. In fact, the literature reveals a number of elaborated and sometime misused definitions for natural interactions [79]. Most of those definitions advocate the use of user's natural abilities, practices, and activities to control interactive systems.

Probably the simplest definition can be devised from Wachs et al. [172] as Voice-based and Kinetic-based Interactions. In the context of our research, Kinetic Interactions are caused and characterized by motion and movement activities, ranging from pointing, clicking, and grasping to walking, balancing or even dancing [10]. There have been some interesting research on movement, as a center anchor to human skills, as part of movement-based interactions as in Hummels et al. [75]. McMahan et. al. [117] defined NI techniques "to be those techniques that mimic real-world interaction by using body movements and actions that are similar to those used for the same task in the physical world". At the sometime, McMahan et. al have actually acknowledged that fidelity for NI to mimic real world actions does vary greatly when comparing NIs amongst each other and with "non-natural" interactions.

Dourish, as reported in [79], argued that the move towards natural interactions and more adoption of human natural skills went through various milestones, starting from exploiting linguistic abilities (as in programming languages), moving to visual memory and special organization (as in Graphical User Interface (GUI)), and utilizing epistemic action (as in tangible user interfaces). Iacolina et al. [79] pointed out that the focus of HCI research is shifting from interface design to interaction design, driven by the spread, use, and goals of ICT technology. This shift highlights fundamental differences between the design from a software perspective to a more social and organizational perspective. In the same paper, NIs are referred to as a "direct manipulation either of physical or virtual objects (such as with tangibles or multitouch displays) accompanied by a narrow class of gestures for disambiguation and negotiation of the interaction space".

Other researchers grounded their definitions of NUIs on the contrast to classical computer interfaces, where such interfaces "employ artificial control devices whose operation has to be learned" [112]. Elsewhere, researchers, such as Norman [126], considered NUI nothing but a marketing term for gesture interfaces.

In his book, Dourish [46] argues that gestures have been a key element to highlight the use of human natural abilities for unobtrusive interaction. Accordingly, gestures have been part of multi-touch interactions [101], whole body interactions [51], multimodal interactions [161], tangible interactions [165], kinetic user interactions [10][24], and embodied interactions [46].

Iacolina et al. [79] argue that defining gestures is not an easy or intuitive task. So far different perspectives have been considered when discussing gestures. Those perspectives vary from an application vs. technology perspective, natural vs. traditional input devices, and personal vs.

collaborative interaction model. Hence, a single definition of gestures is not available in the literature. Moreover, the current landscape of NUI research lacks standardization and standard conventions [126], which we believe are strongly reflected on the variety of available definitions for this paradigm.

2.2.2. What is natural?

As we have seen previously in this chapter, a variety of definitions advocate the natural aspects of NUI. Nonetheless, those natural aspects are highly controversial. Researchers argue that NUIs should go beyond increasing the degree of freedom while performing the interaction, to empowering users for better communication with an interactive system [112]. Others believe that NUIs are not actually natural, as claimed by Norman in on of his recent articles [126].

Additionally, Malizia and Bellucci [112] argue that NUIs impose an "artificial naturality" and a full sense of possible "unobtainable". Despite the employment of more natural body movements those interfaces relay on gestures imposed by designers, which have to be actually explicitly learned. The learning aspects of NUI is a complex issue on its own, as interactions and their corresponding gestures are rarely unified amongst different devices, technologies, cultures, time, and context.

NIs are also criticized of being less accurate compared to other interaction techniques, because of the complexity detection, potential of misinterpretations, confusing commands, and simulation of human body and its movements [119][117]. Precisely for this reasons, some researchers argue that NIs affect the user experience [117], thus affecting the naturalness of those techniques. Others argue that the physicality of interactions enriches the user experience [100]. In the context of this work, we stand by the argument, discussed in [119], that the naturalness of NUI does neither imply the simplicity of learning nor the lack of learning complexity, alternatively the main focus is really on enhancing the engagement of the user's body and its part to create a physically engaging experience.

In the literature, there are limited comparisons of natural and non-natural interactions, leading sometimes to contradictory results. The study of McArthur et al. [115] showed that the former performed significantly worse in terms of throughput, when they compared different pointing techniques based on the Wii controller. On the same stand, a recent study by McMahan et al. [117] reported that some natural interaction techniques (i.e., Remote and Wheel techniques) performed worse than other classical techniques. They reported increasing crash errors and slower lap times (for difficult course), slower lap times (for the easy course), and suffer from latency (action and feedback time). Hence, they have argued that non-natural interaction techniques significantly out performed natural techniques. They also indicated that this can be due to three reasons: the influence of large muscle groups; the mismatch of isotonic sensors for a steering rate control task; and, latency issues. On the contrary, some researchers [117] showed empirically that natural interactions are in fact better and offer greater usability for navigation tasks.

In principle, this research field still requires better understanding and to further develop its design, deployment, and standard conventions [126].

2.2.3. Towards Whole Body Interactions

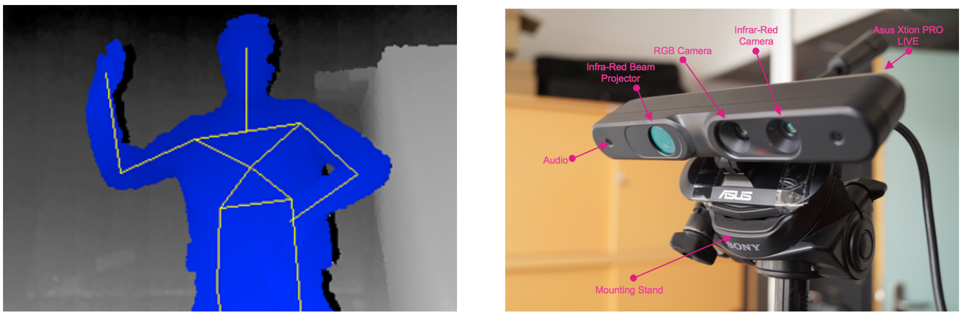
The HCI community is increasingly calling to explore new potential in designing for the whole body in motion [56][51]. This call gains its importance due to an expanding hybrid world of interconnected devices, smart objects, and context-aware services driven by pervasive and ubiquitous computing technologies. Hence, the HCI community is striving to design for the whole body and better inclusion of sensory and motor skills instead of fixing interaction on eyes and hand interactions (as was the case in the mid 80's [25]). Some researchers may differentiate between Whole Body Interactions and movement-based interactions, as the later focus on a specified body part and consist of a pre-defined physical space [119]. Nonetheless, this separation is not widely adopted and hardly present in our work.

One of the closest work in spirit to Whole Body Interactions is Kinesthetic Movement Interaction, which was defined by Moen [118] as "an approach to interaction design which explores free and expressive full body movement as an interaction modality". In fact, Moen is using the two terms exchangeably in a later research work [119]. Fogtmann et al. [56] argue that new conceptual frameworks are needed in order to unveil and explore the new possibilities when designing interactive systems for the full body. Against this background, they have presented a conceptual framework for kinesthetic interactions based on kinesthetic development, kinesthetic means and kinesthetic disorder.

Fogtmann et al. [56] argue that designers and researchers fall short to provide a coherent explanation of the theoretical foundations leading to the design terminology for Whole Body Interactions and Gesture-based Interaction. They also argue that the majority of presented approaches don't explicitly deal with describing the physiological properties of the moving body and the impact of interactions on enhancing the bodily motor skills.

The Whole Body Interaction paradigm is supported with many cutting edge technologies that have been suited for its requirements. First and for most, many interactions relay currently on camera-based technologies that allow for gross and fine detection of body parts and their movements, different body postures, and the body location in space. Many commercial products came to the market in the last four years, including products such as the Microsoft Kinect[5], PrimeSense 3D Sensor (CARMINE)[6], and Asus Xtion PRO LIVE[7] (as seen in Figure 2.5). Moreover a plethora of on-body (i.e., wearable) and environmental sensors came to the market for more fine grained detection of body movements. Probably accelerometer is the most used on-body sensing technology for sensing multiple dimensions of real world actions [154], but other sensors, such as pressure sensors, gyroscopes, and force sensors, are often used as well.

Although the advancement of technology helps to facilitate and lay the ground for Whole Body Interactions, often the research approaches are criticized being technology-driven [119]. Hence, the HCI community continues to explore and investigate other approaches driven by interdisciplinary fields such as fashion design, dance, etc., but still on a much smaller scale [119].



(a) Depth sensor tracking various points on the human body

(b) Asus Xtion PRO LIVE depth sensor

Figure 2.5.: Asus Xtion PRO LIVE depth sensor deployed in our lab

Meanwhile, there are very few conceptual frameworks to design whole body interactions. As mentioned earlier, Fogtmann et al.'s Kinesthetic Framework [56] is one of a few to be found in the literature. The framework is based on three design themes and seven design parameters. The design themes deal with the development (acquiring, developing, or improving bodily skills), means (for reaching a goal), and disorder (experience transformation) aspects of kinesthetic interactions. The design parameters are coined around engagement, sociality (involving other bodies), explicit motivation (interaction involves restricted movements), implicit motivation (involving free movements), movability (body movement restrictions), expressive meaning (the bodily engagement fits the system output), and kinesthetic empathy (when movement patterns are affected by the relation to other people).

[5]http://www.microsoft.com/en-us/kinectforwindows/, latest access on 20.03.2014.
[6]http://www.primesense.com/developers/get-your-sensor/, latest access on 20.03.2014.
[7]http://www.asus.com/Multimedia/Xtion_PRO_LIVE/, latest access on 20.03.2014.

Another framework was proposed by Daiber et al. [42]. They have presented a framework focused on Whole Body interactions with geospatial data. Their framework combines multitouch, foot, and eye gaze input. In fact, the authors claim that this combination is seldom studied in the literature. This approach for geospatial navigation is very interesting, because traditional multitouch based navigation techniques with hand gestures may not be fully adequate for user's interaction with large distance navigation. Instead foot gestures (based on striking metaphor) may provide a good alternative and more suitable interaction possibilities over multitouch in such situations, especially because foot interactions are better suited for continuous input and considered less exhausting [42]. They illustrated the user's interaction with a wall sized mounted display to navigate through a virtual globe. The touch interactions are captured by the multitouch display and the foot gestures are captured by a Wii balance board. Furthermore, foot-based interactions are also considered as an alternative in different situational impairment scenarios [182] (e.g., can be used in cold temperatures as an alternative to hand-based interactions).

Paradoxically, Scoditti et al. [154] pointed out that whilst sensor-based interaction research often present high satisfactory results, they often fail to support designers decisions and researchers analysis. Many questions are still open in terms of manipulation parameters, taxonomies, design spaces, gesture to command mapping, etc. Bailly et al. [15] proposed a set of guidelines for gesture-aware remote controllers based on a series of studies and five novel interaction techniques, but the scope of there guidelines remains limited and is not scalable to other application domains or interaction techniques. Moreover, researchers have pointed out that interactions research still lacks well defined and clear design space and more is required to understand such design space for surface and multitouch gestural interactions [183] and motion-based interactions [146]. Finally, the bodily presence in HCI remains limited due to the subtlety and complexity of of human movement, leaving an open space for further investigations [119].

In the following subsections, we will focus on gestures and movements for interactions as two primary interaction styles facilitating Whole Body interactions. This close highlight aims to further illustrate the potential of human body for interactions but also reflect the challenges facing this area of research.

Gestures for interactions

Gestural interactions are defined, discussed, and classified based on various perspectives. The major part of human gesture classification research is focused on human discourse [183], the other part adopts a dialog approach between human and an interactive device [146]. In fact, gestures are also classified by other dimensions, most commonly by the underlying used input device properties or sensing technology. A rich outline of input device properties for gesture classification was depicted by Hinckley [71]. The classification dimensions include property sensed (linear position, motion, force, rotary devices sense angle, change in angle, and torque), number of dimensions, indirect versus direct, and device acquisition time. Furthermore, other metrics also exist to include gain, speed and accuracy, error rates, learning time, footprint, user preference, comfort, and cost. The diversity of perspectives, definitions, and classifications on gestures has been reflected on wide range and diverse gesture taxonomies such as: Scoditti et al. [154] (focused on gestures interactions based on accelerometers), Ruiz et al. [146] (focused on user defined gestures and illustrated), Karam et al. [87][88] (one of the most intensive surveys covering about 40 years of gesture research as illustrated in Figure 2.6), and Alaoui et al. [3] (focused on event-based and continuous gestures).

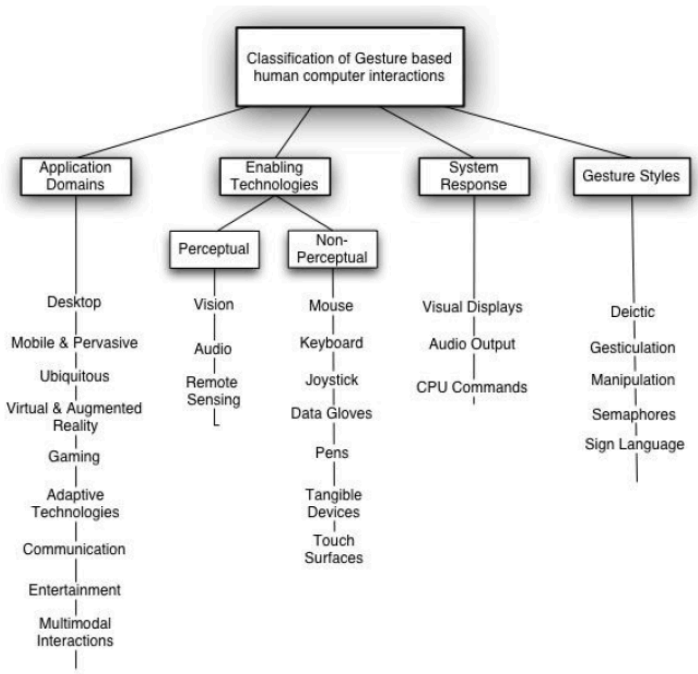
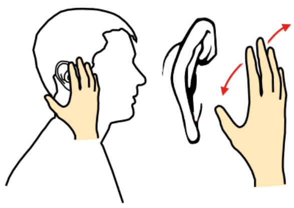


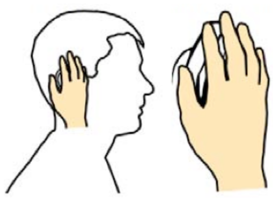
Figure 2.6.: Gesture taxonomy based on application domain, technology, response, and style (from [88], used with permissions from the author)

Out of the various classifications covered in Figure 2.6, the style classification is briefly discussed in this section. Other classifications covered in the figure are rather self explanatory. According to [88] gesture styles are often combined together and gestures are not often composed from a single style. Karam and Schraefel [88] classified gesture styles into the following five categories: Deictic gestures which establish an object identity or location by pointing as in desktop computer [96][121], object manipulation, or targeting appliances [58]; Manipulative gestures which allow to control an entity by applying tight relationship between gesture movements and a manipulation action (e.g., for instance, manipulations of physical objects as in virtual reality interfaces, drag and drop data between devices virtually in the space, manipulation robot arms [164], and many tangible computing interfaces); Semaphoric gestures employ a stylized dictionary of static or dynamic hand or arm gestures (e.g., gesturing with the thumb and forefinger to represent the "ok" symbol); Gesticulations which rely on analyzing body movements (focused on hand movements in many research projects) within the context of user's speech; and Language gestures which relate to sign language and is performed based on a formal and well defined grammatical conversational structures.

Principally, gestures may be also classified by combining multiple classification dimensions. For instance, classifications based on used technology and number of dimensions as in two-dimensional surface gestures and three-dimensional motion gestures [146]. Different examples of motion, touch, and combination gestures are presented in Figure 2.7, Figure 2.8, and Figure 2.9 respectively. The classification of gestures to motion and touch gestures is perhaps the most closely relevant classification to the goals and aims of this dissertation.

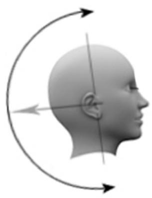


(a) Mid-air gesture (from [103])

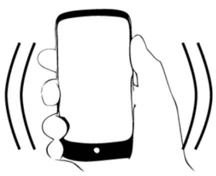


(b) Cover ear gesture (from [103])

(c) Full body rotation gesture (from [8])

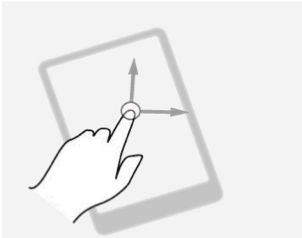


(d) Head gesture (from [113])

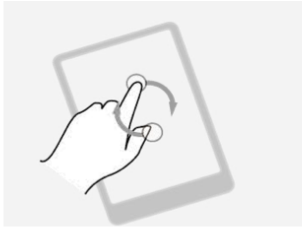


(e) Shake gesture (from [146])

Figure 2.7.: Motion gesture examples (all figures are used with permissions from the corresponding authors)

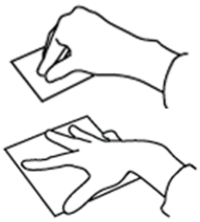


(a) One finger drag gesture (from [89], used with permissions from the author)



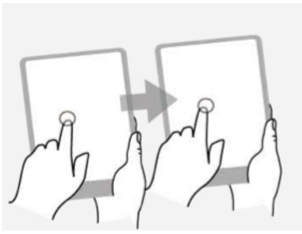
(b) Pinch in / out gesture (from [89], used with permissions from the author)

(c) Scratch out gesture (from [183], Image courtesy of Jacob O. Wobbrock)

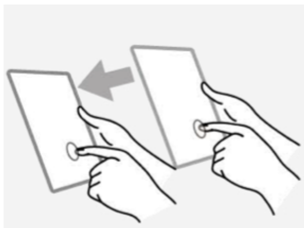


(d) Splay fingers gesture (from [183], Image courtesy of Jacob O. Wobbrock)

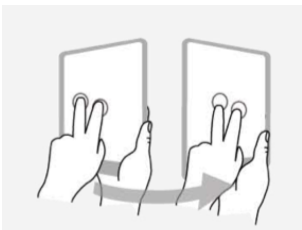
Figure 2.8.: Touch gesture examples



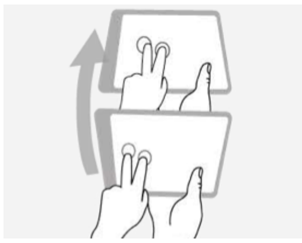
(a) Transitional shift gesture



(b) Vertical shift gesture



(c) Yaw rotation gesture



(d) Pitch rotation gesture

Figure 2.9.: Combined gesture examples (from [89], used with permissions from the author)

Movements for interactions

Several research fields have been undertaken to develop techniques and technologies for utilizing human body movements as input for interactive systems. In the previous two sections, we have covered different aspects of the gesture and whole-body interaction research. While overlap exists to the previously mentioned sections, the focus herein is on movement as an element for interactions.

The research on utilizing movements for interaction is spread over a wide research landscape. Computer vision studies different approaches to visually analyze and recognize human motion on multiple levels (i.e., body parts, whole body, and high level human activities) [105]. Other research projects involve affective computing to study expressive movements as in the EMOTE model [35] and EyesWeb [28], visual analysis [2] and representation [14] of movements, etc. The literature is rich with examples on movement for interactions. Rekimoto [140] presented one of the earliest work for mapping motion (e.g., tilting) to navigate menus, interact with scroll bars, pan, zoom, and to perform manipulate actions on 3D objects. The research effort on tilting was then followed, especially in the mobile interaction area by Harrison et al. [68] and Bartlett [17]. Meanwhile, Hinckley et al.'s [72] idea of using tilting for controlling the mobile screen orientation is one of the most widely adopted techniques implemented in many mobile phones currently sold on the market.

In their work on movement-based interactions, Loke et al. [105] presented an interesting analysis on the design of movement-based interactions from four different frameworks and perspectives: Suchman's framework covered the communicative resources for interacting humans and machines; Benford et al.'s framework (based on Expected, Sensed and Desired movements) for designing sensing-based Interactions; Bellotti et al.'s framework (Address, Attention, Action, Alignment, Accident) for sensor-based systems; and, Labanotation as one of the most popular systems of analyzing and recording movement. In Benford et al.'s framework "Expected" movements are the natural movements that users do, "Sensed" movements those which can be sensed by an interactive system, "Desired" movements are those which assemble commands for a particular applications. In Bellotti et al.'s framework "Address" refers to the communication with an interactive system, "Attention" indicates whether the system is attending to the user, "Action" defines the interaction goal for the system, "Alignment" refers to monitoring the system response, and finally "Accident" refers to errors avoidance and recovery.

The richness of human body movements makes human movement an overwhelming subject for interaction design. The hand and its movements, for instance, provide an open list of interaction possibilities. In his work, Mulder [120] listed just a subset of hand movements that reflects interaction possibilities, which included: accusation (index pointing); moving objects; touching objects; manipulating objects; waving and saluting; pointing to real and abstract objects; and positioning objects. Moreover, he described and categorized hand movements into goal directed manipulation, empty-handed gestures, and haptic exploration. This classification reveals the potential of one individual part of human body. The goal directed manipulation category includes movement for changing position (e.g., lift, move, thrust, shake, shove, shift, etc.), changing orientation (e.g., turn, spin, rotate, revolve, twist), changing shape (squeeze, pinch, wrench, extend, twitch, etc), contact with the object (i.g., grasp, grab, catch, grip, hold, snatch, clutch, etc.), joining objects (e.g., tie, pinion, nail, sew, etc.), and indirect manipulation (set, strop). The empty-handed gestures category included examples such as twiddle, wave, snap, point, etc. Finally, the haptic exploration category included touch, stroke, strum, thrum, twang, etc. In the same work, he also indicated that there are other types of categorization based on communication aspects for example. Yet, this potential grows greatly when considering the rich nature of natural interaction techniques, as in Whole Body Interactions for instance.

The notion of movement qualities is well studied and applied in different fields, especially in dance and choreography. Despite the importance of movement for interaction, the HCI field does not yet explore this notion. In fact, the primary foundations of movement qualities are very poorly discussed in the HCI literature [3], despite some recent research contributions as James et al. (interactions technique based on dance performance) [81], Moen (applying Laban effort dance theory for designing of movement-based interaction) [119], Alaoui et al. (movement qualities as interaction modalities) [3], and Hashim et al. (movement analysis for graceful interaction) [69].

In the previous section (Section 2.2.3) we have presented various research examples on gestures as one of the most wide spread research effort to utilize movement for interactions. Herein, we focus on the explicit consideration and use of movement qualities for interaction techniques. Two of the rare examples of using pure movement qualities for an interaction techniques were reported by [119] and [3] as part of two prototypes called the BodyBug and the Light Touch respectively. BodyBug is a wearable interactive box linked with a wire and react to the users movements (based on Laban effort model). In the Light Touch prototype three movement qualities (i.e., Breathing, Expanding, and Reducing) are used to control a light installation. Every property corresponded to a dedicated action, for instance "Breathing" corresponds to the blinking of the light. Additionally, the authors presented a more general conceptual framework for movement qualities for interactions [3].

In his work, Moen [119] discussed different aspects of human movements and their association with different design criteria, as listed in Table 2.1, including personal space, natural movements, movement impulses, movement impression and expression, and movement as fun.

Movement Aspect	Movement-Based Design Criteria
The personal interaction space	• Three dimensional interaction space • Mobile user-defined interaction space • Tangible interaction near the body • Independency of visual or audio output
Natural movements	• Support free • explorative movements • Support individual preferences
Movement impulses	• Create movement that trigger movement • Use the kinesthetic sense • No specified use or punishments are given
Movement as impression and expression	• All kinds of movements make sense in relation to possible input/output • Individually and collaboratively use • Movement dialogue
Movement is fun!	• Movement for the sake of movement

Table 2.1.: Human movement aspects and design criteria (from [119])

To our best knowledge, universal design guidelines for movement-based interactions are not easily found in the literature. Recently, Gerling et al. [62] proposed seven guidelines for Whole Body Interactions with games. Although created based on gaming scenarios and focused on elderly population, the guidelines can be generalized to some extend. Based on Gerling et al.'s guidelines we have adopted and reframed the most relevant and universally applicable guidelines:

- **Age-Inclusive Design** where age-related physical and cognitive impairments are acknowledged and considered;

- **Adaptability as part of the interaction** where individual user characteristics (e.g., range of motion) are considered as part of an adaptive interaction;

- **Exertion Management** where physical effort is managed to prevent physical fatigue and overexertion;

- **Manageable Interactions** where natural mapping and empowering interaction recall are applied; and

- **Continuos Support** where enough material and tutorial are provided for users to perform the interaction correctly.

Similar to the previously discussed projects, we place human body movements as the central focal point in designing and executing interaction for NUIs. Nonetheless, this dissertation focuses on composing interaction possibilities form multiple interaction techniques that relay on multiple body parts and their movements. This position put human body movement at the core of the dissertation conceptual design (appears in Chapter 3) and the implementation of movement profiles (appears in Chapter 6). We view interaction techniques as a subpart of a united body of interactions that adapt to the user's physical context and needs.

2.2.4. Social acceptability of NUI

In section 2.1.4, the general social aspects in designing AAL were discussed, in this section, the social acceptability of NUI is highlighted. The major technological advancement and wide spread adoption of NUI, as we have covered earlier in section 2.2.3, have triggered the demand for various acceptability investigations. Particularly, the diversity of gesture definitions is reflected on the diversity of NUI based interactive systems, hence resulting in a wide range of interaction forms and usability [141]. Along side the development of new interaction techniques, researchers have been recently investigating the social acceptability of NUI. Various research papers have indicated that using mid-air gestures in domestic scenarios are well accepted by users, as well as gestures offer several advantages [91][141][15].

In fact, Norman [126] argues that touch and motion based interactive systems are widely accepted by users. This acceptance is reflected on the users' natural interaction behaviour in different domestic scenarios. Due to this wide acceptance, users are becoming more acquainted with ambient interactive systems and expect widely covered intelligent behaviour and response in various physical spaces. Nonetheless, due to the currently limited deployment scale of AmI and the absence of widely spread interactivity, users often experience miss match between expected behaviours in the physical world, such as tapping on non-touch surfaces, waving hands to dispense water in front of sinks with no infrared sensors, etc. Norman argues that those behaviours are becoming part of the users' natural expectation of interactivity with physical environments.

An interesting study, reported by Gerling et al. [62], was conducted on elderly adults and demonstrated the benefits of Whole Body Interactions in gaming. The study revealed that this type of interaction may positively increase the users' mood and their emotional well-being, hence increasing the users' acceptance.

It is important to acknowledge that the acceptability of NUI is dependent on the type of performed interaction and the context of execution (e.g., the environment). A study by Rico and Brewster [141] demonstrated a strong correlation between gesture types and the surrounding environment (i.e., private settings, public settings, and semi-public settings) on the gesture acceptability. Increasing privacy levels results into a higher acceptability and willingness to perform gestures. Moreover, the study indicated that gestures with visual clues to the surrounding audiences are more likely to be used than those gestures lacking visual clues. Moreover, their results revealed that the most acceptable gestures show the following characteristics: (1) Require subtle movements such as tapping; (2) Similar to everyday actions such as shaking; and, (3) Enjoyable movements. On the other hand, the least acceptable gestures are characterized as: (1) Weird looking, for instance shoulder and nose taps are considered quite attention seeking; (2) Physically uncomfortable such as head nodding and foot tapping; (3) Interferes with communication, for example head nodding may be confusing during the conversation; and, finally (4) Uncommon movement.

To increase the acceptability of NUI, new types of guessablity studies aim at end-user election of NUI. For instance, Wobbrock et al. [183] presented an extensive set of user-elected touch gestures and a study by Ruiz et al. [146] presented another set for motion gestures. The later study demonstrated that 82% of the study responses indicate the acceptability of using motion gestures at least occasionally. Those studies are particularly interesting for our research because they invoked and demonstrated the importance of deeper understanding of user needs and preferences for designing NUI.

2.2.5. The physical abilities heterogeneity challenge

Meanwhile, as we have discussed in section 2.1, the fast growing population of elderly is becoming a crucial demographical shift to consider for HCI research [182]. Nonetheless, this demographic shift is only a manifestation of the increasing diversity of users' physical capabilities caused by age, gender, culture, social context, etc. In the year 2010, the EU ICT strategy [159]

Section 6.5 covers various health related terms and definitions are covered in Table 6.1 according to WHO called for designing user-friendly interfaces to support devices everywhere. This explicitly called for considering the diversity of the user population and the importance of inclusion, especially for elderly people and physically challenged people. Hence, the users' needs, abilities, and impairments should be prime focal points in ambient spaces and their interactive eco-systems.

Life expectancy is adversely affected by decreasing physical and cognitive activities, often leading to sedentary death syndrome [62]. Gerling et al. [62], Piper et al. [134], and Wobbrock [182] discussed some interesting age-related challenges of great negative impact on NUI, which can by equally caused by other reasons such as accidents and illness. First, human sensing, attention, and memory are all subject to great changes due to age. Second, people face decrease in muscle strength, Third, age forces a significantly reduced movement control, constrained movement coordination, and longer movement execution time. Fourth, elderly face noticeable decrease in balance and gait. Fifth, age causes elderly to suffer from increased motor learning difficulties. Sixth, age related and cardiovascular diseases, such as arthritis and stroke respectively, cause a great negative impact on human body movements. This wide range of disabilities and impairments creates a great tension on interactive eco-systems and reduces their usage potential, if not dealt with and handled correctly. For instance, reduced physical abilities adversely reduce the ability to interact with multitouch surfaces [134], hence suggesting to avoid fine motor input.

Two of the four suggested trends of designing interactions on mobile devices suggested in [182] were related directly to the disability challenge and the variation of physical abilities. First, designing to improve accessibility for a growing aging population; second, responding to situational impairments [155] [156]. The different physical impairments are categorized by Sears [156] to four main categories: Structural deviations are identified by a significant deviation or loss of a body part; Mobility addresses issues of joint and bone movements; Muscle power functions address a partial or total loss of muscle power for contractions; and finally, motor functions address the ability to control voluntary and involuntary movements.

Paradoxically, whilst NUI strive to engage a better bodily experience and counter balance the users' physical limitations, this new era of NUI is greatly challenged by the impact of wide range disabilities and heterogeneity of physical context on the quality and ability to perform interactions. This calls for interactions to be adaptive and assistive to increase (or at least maintain) the functional capabilities of individuals to interact with ambient spaces. The main goal is to normalize the adoption of interaction techniques. Normalization, defined and discussed in [187], is achieved by allowing users to act and perform "normally", regardless of their physical conditions.

The design of ambient spaces and interactive eco-systems should be inclusive (universal design or design-for-all) to include users of different physical, cognitive, or social conditions [187][182], hence the variation of physical abilities should not be made as a barrier to correctly interact with ambient spaces. Yee [187] suggested different aspects to be considered for selecting input devices, which we think can be generalized as important context characteristics for selecting and utilizing interaction techniques in ambient spaces including: physical capabilities, controlled voluntary movement (e.g., fine motor control, range of motion, strength, fatigue, and multiple movements), cognitive capabilities, sensorial capabilities, personal considerations, environmental conditions, tasks to carry out, temporal considerations, financial considerations, portability considerations, and normalization criterion.

Interaction, disability, and aging where the subjects of various research, for instance SINA [187] as a hands-free interface based on computer vision techniques for motion impaired users, inclusive mobile phone design for elderly [133], surface computing accessibility for elderly [134][108], comparing touch surfaces and traditional input devices (i.e., mouse, trackball and stylus) for users with physical impairments [21], comparing input devices for elderly [82], physical impairments and interaction adaptation [19][59][171][184], analysis on touchscreen use by users with motor impairments [12], novel interaction techniques for users with disabilities [76][78][163], etc.

Many of those studies nonetheless point out that investigating interactions for elderly population is still relatively weakly investigated. The research focus is often targeted at the general public or largely limited to lab studies, as in surface computing for example [134] [12]. Anthony et al. [12] reported, in a recent study, that the accessibility of surface computing is better explored for visually impaired users than for physical motor disabilities. A number of general design observations, on the disability challenges in ambient spaces, can be also drawn from the studies cited above:

- **Recognizing similarities and differences**: Similar expectations from interactions amongst different age groups may exist, as well as the expected differences. A direct comparison between older and younger adults, by conducting an experiment on a multitouch surface, indicated that the groups' interaction expectation were similar, but differences in the process of discovering interaction possibilities exist. Younger adults showed faster patterns to try various ways for interactions. As expected, older adults also showed slower interaction execution time and higher hesitancy [134].

- **Technology should improve to provide higher adaptation possibilities**: Anthony et al. [12] reported that there is a strong indication that the technology should still be improved to cover the physical needs of users. This should range from tailored interaction styles to novel indirect alternative methods.

- **Basic interaction functions are challenging for users with disabilities**: This challenge calls for alternative support mechanisms for impairments are highly commendable [12].

2.2.6. NUI fusion and adaptation

Parts of this section appear in [10]

Reviewing the literature reveals an extensive effort in the area of traditional user interfaces adaptation in terms of context modeling, user modeling, automatic generation of interfaces, etc. One of the well-established concepts is plasticity [26]. It refers to the capacity of an interactive system to tolerate changes in the context of use while retaining usability. Plasticity relies on adapting the GUI according to three factors: input, output, and platform. The WWHT framework [143], on the other hand, is based on a rule-based system, which matches different communication channels to a given context model, based on four levels of adaptation (What, Which, How, Then). Moreover, most available adaptation approaches are not considered adequate for the natural characteristics of ambient environments [137] as pointed out previously in section 1.2, particularly due to heterogeneity and distributivity, dynamic media mobility, and user mobility.

In addition, most adaptation approaches focus on interface issues such as information presentation but not the interaction per se. Pruvost et al. [137] clearly indicated that locking interaction devices in their own closed world is certainly an issue for interaction systems adaptability in ambient spaces. This closeness results into reducing the richness and unity of those interaction devices in various context scenarios.

Pruvost et al. [137] also argued for highly adaptable user interfaces that preserve utility and usability across contexts. In their described adaptation vision, they have presented the concept of Off-the-shelf Interaction Objects, which are pre-implemented bundles of code, intended to be reused and composed at runtime. The objects aim to provide the necessary adaptation required for the interaction technique. While their vision is focused on the structural adaptation of user interfaces and the adaptation of a running dialogue, our work is more concerned with the documentation, sharing, and deployment aspects of NUI, especially kinetic interactions.

The Gestureworks Core[8], which is limited to multitouch interactions, is one of the earliest multitouch gesture authoring solution for touch-enabled devices on a variety of platforms such as Flash; C++; Java; .NET; Python; and Unity. Based on the Gesture Markup Language (GML), the solution comes with a rich library of pre-built gestures and allows for new custom gestures and gesture sequences to be built by designers. The OpenNI[9] is an

[8] http://gestureworks.com, accessed on 23.03.2014.
[9] http://www.openni.org, accessed on 23.03.2014.

open source SDK used for the development of 3D sensing applications and middleware libraries. The main targets of this framework include enhancing the NI techniques development community; making it possible for developers to share ideas; to share code with each other; and to address the complete development lifecycle by a standard 3D sensing framework.

In contrast to whole gesture recognition, the catchment feature model [138] applies a feature decomposition approach for gesture component fusion. Accordingly, gesture analysis is based on gesture features cohesion, segmentation, and recurrence. Nonetheless, this type of fusion targets micro-level fusion amongst a single interaction gesture.

Despite the HCI innovations in the last two decades, the fusion and composition of NI techniques into ensembles of interaction techniques are still rather unexplored areas. This is due to various reasons including: combined input resources have gained only little attention, especially in the case of hands and foot interaction resources [42]; much of the literature focuses on using a limited part of the body for interaction; limited investigations have covered the gross motor skills and utilized the kinesthetic sense [56]. Few recent HCI research calls, such as Pruvost et al. [137], have argued for interaction environments to be open and dynamic, due to the complexity of interaction contexts.

Interaction fusion and composition for NUIs are becoming essential needs for ambient spaces. To the best of our knowledge, the research effort to tackle these problems remains very limited and our work is one of few early research efforts in these directions.

The conceptual approach towards interaction composition is presented in section 3.2

2.3. Outlook

In this chapter, we have presented an literature review on related topics about AAL and NUI in ambient spaces. We have identified the wide research effort and advancements in those fields. This effort is manifested by a large number of AmI, AAL, and NUI projects, only some of which are discussed in the chapter. Clearly, the diversity of the user population in terms of age and physical abilities imposes various emerging technical and social challenges for designing and deploying of NUIs, specially with increasing application domains relaying on AAL technologies. This vast expansion of AAL has lead to various investigations on its social implications. The increasing complexity of controlling, managing, and interacting with AmI technologies are strong reasons for hindering users in most AAL scenarios. The contributions of this dissertation are part of the community effort to tackle one of those challenges, namely the interaction challenge.

Various sections of this chapter clearly demonstrated that studying this area of research is challenging because of the absence of well define agreed upon definitions and the existing various view points, classifications, and taxonomies involved in studying the NUI paradigm. Moreover, the lack of universal design guidelines for movement-based interactions increases the difficulty of tackling the design of ambient interactive systems.

Paradoxically, whilst NUI strive to engage a better bodily experience and counter balance the users' physical limitations, this new era of NUI is greatly challenged by the impact of a wide range disabilities, which impact the user's ability and quality to perform interactions. This calls for interactions to be adaptive and assistive to increase (or at least maintain) the functional capabilities of individuals with disabilities to interact with ambient spaces. While various research activities have been actively carried on, it is still clearly visible that interaction innovation is targeted at limited parts of the human body, leaving parts of the body (e.g., hands) much more studied and investigated than other parts (e.g., foot or neck). This dissertation strives to investigate the whole-body motion potential in ambient spaces, as part of a recently targeted goal aimed at by the HCI community. Principally, this dissertation focuses on composing interaction possibilities from multiple interaction techniques that relay on multiple body parts and their movements. This position put human body movement at the core of the dissertation theoretical concept (appears in Chapter 3) and the implementation of movement profiles (appears in Chapter 6). We view an interaction techniques as a subpart of a united body of interactions that adapt to the user's physical context and needs. The core concept of this dissertation is introduced in section 1.4 and a lengthly discussion follows in sections 3.2 to 3.4, based on the novel idea of dynamic ensemble of interaction techniques that maximize the whole body experience and physical integration of the users' physical bodily abilities.

3

Approach: Anthropometric Framework for Natural Interaction Ensembles

Parts of this chapter have been published in [10] and [5]

In section 1.4, we have identified that interactions with NUIs, especially kinetic interactions, are currently hindered due to the lack of systematic consideration of the increasing diversity of user populations in ambient spaces; and effective means for documenting, adapting, and deploying interaction techniques. It is argued in this dissertation that interactive eco-systems should enforce better performance and integration of users within their known physical abilities.

In this chapter, the concept of *Interaction Ensembles* is presented as the core approach of this dissertation. *Interaction Ensembles* is a novel approach for adapting, sharing, and runtime deploying of natural interaction techniques in ambient spaces. This chapter aims to present and discuss the concept and its essential building blocks. The technical realization of this approach is exploited by the STAGE framework, which is extensively presented in Chapter 4 and Chapter 6.

In section 3.1, the chapter starts by briefly reflecting on one of the most applied design approaches, namely the "one-design-fits-all" approach and presents two of its main restrictions. Section 3.2, we argue for de-coupling interactions in ambient spaces. Section 3.3 introduces relevant concepts to the theoretical concept of this thesis including input tasks and interaction primitives. Shortly after, section 3.4 presents the core definitions of interaction ensemble and interaction plugin. To further elaborate those definitions, a full walkthrough user scenario is presented in section 3.5 and six different interaction ensemble cases are presented in section 3.6. Building and orchestrating Interaction Ensembles as theoretical concept is further presented in section 3.7. The section is split into four subsections discussing; first, building Interaction Plugins (3.7.1); second, anthropometric driven matching and presentation of Interaction Plugins (3.7.2); third, on demand wiring of interaction resources (3.7.3); fourth, community-based designing and sharing of IPs (3.7.4). Finally, the chapter is conceded by an outlook section (3.8).

3.1. The "one-design-fits-all" approach

User diversity poses great challenges on interactions with NUI as introduced in section 2.2.5. While the majority of commodity devices target the average user, a lot of dedicated devices are specifically manufactured to support people with special physical needs. Examples are oversized trackball mouse and adaptive keyboard devices for users suffering from non-reliable muscle control and lack of precise movements [176].

Kane et al. [86] studied mobile interactions with motor-impaired people. The authors concluded that a clear mismatch between the available devices and abilities of the motor-impaired participants exists, since none of impaired participants used accessibility-enabled mobile devices. Various issues and observations were evoked in this study. One participant reported a successful use of an accessibility keyboard designed for children. She adopted the keyboard for interaction with her home personal computer. However, she rejected the use of accessibility devices in public. Another participant conveyed some privacy issues with using a portable magnifier in public, which is used otherwise to interact with her phone screen in private settings.

Obviously, designing interactions for NUI devices is challenging, because it does not often fully explore the potential of motor interaction, even when optimized for the considered impairments. This is mainly due to the following restrictions:

- First, these devices are usually designed with a specific impairment in mind but still compromising the variation of degrees of disabilities.

- Second, these devices are usually context-agnostic; resulting in one-design-fits-all approach.

Interaction devices and interaction techniques should enable seamless users' interactions with no prior restrictions, complying with the "come as you are" requirement. HCI systems should not pose requirements on the user such as wearing dedicated markers or cloths, or fixing the background [172]. Similarly, the user should be able to interact with the system based on her current physical qualities and abilities without making interactions cumbersome.

In the rest of this chapter, an alternative and novel approach will be presented and discussed.

3.2. Towards de-coupling interactions in ambient spaces

The conventional static binding approaches between devices, interaction techniques, and applications offer various advantages such as good predictability, stability, and ensured availability. Despite these advantages, the approach is greatly challenged in the context of interactions in ambient spaces. Ambient spaces impose high level of user mobility, diversity and heterogeneity of context (i.e., user, environmental, and social), and dynamically changing resources (i.e., continuously changing environment as devices join and leave at any time).

In this dissertation, a novel approach to de-couple the close and static binding between devices, interaction techniques, and applications is suggested. Alternatively, the approach aims to utilize dynamic compositions of NIs, assembled and configured based on user's capabilities and situational context, in an ad-hoc manner.

Hence, our approach:

- **fosters soft-wired applications and devices**, in order to overcome the limitations of the static binding and to address one of the most challenging requirements in pervasive environments, namely the "come as you are" requirement.

- **tackles the mismatch problems between users' needs and devices' offers**, which can be avoided by employing the best matching NIs to the given context, hence the user independence (acceptability by permitting customizability) and usability qualities required by Wachs et al. [172] are inherently enhanced.

- **matches also calls form the HCI community to overcome challenging issues for user interfaces in ambient spaces including environment heterogeneity, context complexity, and increasing adaptability**. Pruvost et al. [137] noted that interaction environments are becoming increasingly heterogeneous and dynamic, hence they are no longer static and closed; the interaction context is becoming increasingly more complex; and increasing adaptability is required for sustainable utility and usability.

The combination of multiple interactions is not well studied in the literature. Obvious combinations such as hand-based and foot-based interactions have gained little attention [151] and have been investigated in few studies as in [42][151]. The combination of different interaction modalities may improve the user experience and reduce ergonomic problems (e.g., arm fatigue). For example, navigating a large geographical map with multitouch wiping gesture may lead to arm fatigue [42]. Integrating foot interaction (for example, utilizing weight for navigation as in striking metaphor) may be more adequate and intuitive in this case.

As introduced earlier in section 1.4, this work envisions and strives to establish a theoretical concept and achieve a framework for anthropometric driven ensembles of NI techniques. Interaction modalities are tailored at runtime to maximize the adoption of interactive systems according to the users' physical abilities, needs, and context. This is based on the theoretical concept of utilizing detailed anthropometric data and human ability profiles for maximizing the usability of kinetic-based NI for acting on the stage of an ambient environment.

3.3. Interaction primitives and interaction design space

Foley et al. [57] presented a taxonomy for input devices based on six graphical subtasks, called Interaction Tasks, required in any graphic-based application: Position (specifying a 2D or 3D position), Select (choosing an element from a set of choices), Path (specifying a number of positions over a specific time), Orient (rotating an object), Quantify (specifying a numeric value), and Text entry (entering character strings). The cross product of interaction tasks with input devices shows the different ways each interaction task can be performed with existing devices. Nonetheless, this taxonomy was criticized by the same authors. They clearly noted that the task of creating a complete list is not possible, as novel interaction possibilities cannot be anticipated. The taxonomy was also criticized by other researchers of being not adequately generic. Further limitations and shortcomings of this taxonomy were extensively discussed in [29].

To provide an acceptable level of generality, morphological taxonomies as in Mackinlay et al. [110], propose a classification based on the physical properties sensed by input devices. Mackinlay et al. argued that such a morphological taxonomy provides a wide and flexible range of generality to describe any input devices. Most importantly, this classification includes the physical nature of the action and goes beyond graphical user interfaces.

In the context of interaction in ambient spaces, it is argued that the two aforementioned taxonomy types are not adequate [16]. From one side Foley et al.'s taxonomy straggles to go beyond GUI and, on the other side, Mackinlay et al.'s taxonomy fails to describe multimodal devices.

Of a particular relevance to this dissertation, Scoditti et al. [154] proposed one of very few taxonomies for gestures, which can be applied in ambient spaces. The new taxonomy was based on four main principles: (1) Semantic, syntactic, lexical, and pragmatic issues of interaction; (2) User centered perspective, i.e., physical human actions are premium; (3) Context; and (4) Foreground and background interaction. To validate the proposed taxonomy, they classified more than 30 research projects in this domain. Interestingly, their validation revealed the dominance of the Select and Position interaction tasks, while Path (presenting the largest set of gestures in NUI) is the least explored.

Although our work is based on the existing previously mentioned works, it adopts the semantics of users' physical actions in the ambient space, rather than the device physical properties or the graphical subtasks. Hence, we particularly re-use the semantics of Foley et al.'s GUI interaction tasks as elementary (or basic) input tasks in ambient spaces. Input tasks describe the semantics of the physical actions that should be performed by the user, regardless of the input device or modality used for acquisition. The basic input tasks are: Position; Select; Path; Orient; Modification (alternative to quantification, indicating a change of a physical property or the shape of an entity); and Semiotic (equivalent to Text, e.g., writing voice-based tasks, code scanning, writing tasks, etc).

The concept of interaction primitives for NUI is generally not well defined in the literature, especially for ambient spaces. We refer to interaction primitives as the basic interaction units that glue between physical input devices and representations consumed by applications [10]. This notion of interaction primitives is similar to van der Vlist et al. [166] and Niezen [125]. They defined interaction primitives as: "the smallest addressable element that has a meaningful relation to the interaction itself". In fact, due to the lack of comprehensive input taxonomies for ambient spaces, Niezen's ontology adopted Mackinlay et al.'s suggested domain set [110] despite the shortcomings of this taxonomy as discussed earlier in this section.

Interaction primitives are either directly consumed by applications or translated to a particular representation according to the application. There have been few studies that focus on interaction primitives for dedicated application domains. Daiber et al. [42] investigated interaction primitives for whole body interaction with geospatial data. In this application domain, pan, zoom, and selection interaction primitive are the most typical. This effort had been initially limited to a combination of multitouch and foot-based gestures, but was later extended to a framework for multitouch, foot, and eye gaze input. For multitouch surfaces, Piper et al. [134] observed the accessibility of eight interaction primitives: select, move, resize, rotate, pan, draw, read text, and enter text.

We extend Niezen's semantic ontology [125] with the concept of interaction plugins. In Figure 3.1, colored nodes pronounce the differences to Niezen's ontology. Principally, the main proposed differences are caused by

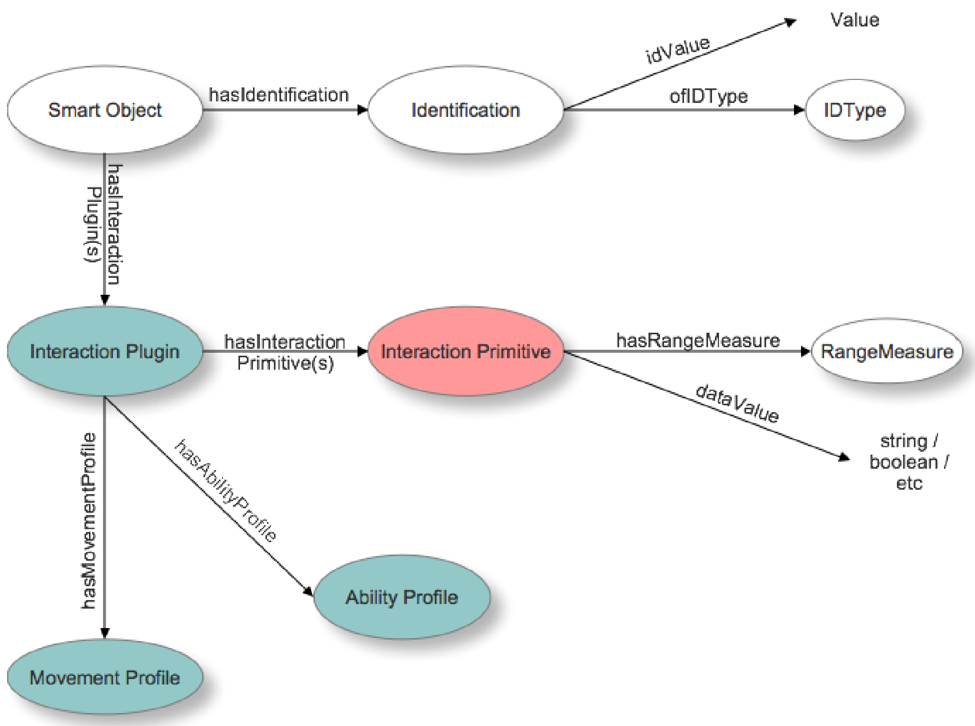


Figure 3.1.: Semantic Interaction Ontology in Ambient Spaces: This ontology is an extended and modified version of the Semantic Interaction Ontology from [166][125] (Colored nodes: indicate either newly proposed or modified nodes; White nodes: refer to the original ontology)

introducing the concept of interaction plugin, which highlights the interaction capabilities rather than the device and device's raw input modalities in the original ontology.

Movement profiles (section 6.4) describe the required movement for interactions and ability profiles (section 6.5) describe the required physical qualities for interactions.

Niezen's ontology treats objects in ambient spaces as smart objects, each is identified by a unique ID and has a value and a type. More importantly, smart objects contain various interaction plugins. An interaction plugin has various interaction primitives, an ability profile, and a movement profile. The two prime properties of an interaction primitive is the range measure and data value. The range measure according to Niezen is adopted from Mackinlay's domain set, which includes the transformations required to map the input to various interaction primitives or events. The data value describes the actual value of an interaction primitive.

3.4. Interaction Plugins and Interaction Ensembles

Principally, the theoretical concept suggested in this thesis treats each NI technique as a standalone interaction interpreter entity, called *Interaction Plugin*, which is defined in the context of this dissertation as follows:

> Definition - **Interaction Plugin**:
>
> An executable component in ambient interactive systems that encapsulates a single natural interaction technique with a set of elementary input tasks as input and delivers higher level interaction primitives to applications based on specific interaction semantics.

Interaction plugins allow NI techniques to be discoverable, exportable, exchangeable, plug-able, and sharable. In the definition, elementary input tasks are considered the unit of an entry of information by the user. Interaction primitives are considered the basic interaction units that glue between physical input devices and interaction.

Elementary Input Tasks and Interaction Primitives are defined in Section 3.3

The second building block of our approach is the concept of *Interaction Ensemble*, which is defined as follows:

> Definition - **Interaction Ensemble**:
>
> Multiple interaction techniques (i.e., interaction plugins) are tailored at runtime to adapt the available interaction resources and possibilities to the user's physical abilities, needs, and context.

For multitouch surfaces, Piper et al. [134] investigated the accessibility of eight interaction primitives, out of which the "selecting" interaction primitive was the most straight-forward and physically manageable for participants. Other interaction primitives were more challenging but still could be performed successfully. For example, the "moving" objects for participants with hand tremors. The "Panning" interaction primitives was very easy to perform but required the longest time to figure out the action required.

3.5. User scenario - smart rehabilitation center

User scenarios are one of the popular interaction and software design methods to aid the design processes and reveal requirements. In this section we present one user scenario used to aid investigating the concept of interaction ensembles:

This scenario is used through out the dissertation as a coherent illustrative exemplification of the different discussed concepts.

> John is admitted to a Rehabilitation Residential Center for two weeks. In this time period, he will undergo various intensive rehabilitation and monitoring activities. John is 35 years old and works as an editor for a regional ICT magazine. He is currently suffering from movements rigidity and limited range of motion in both of his hands, due to a recent car accident. John got to know Aimee, one of the center's residents, in one of the social activities. She is a 28 years old high-school math teacher. She is suffering from a balance disorder due to a sudden deterioration in her vestibular system. In very short time, Aimee and John became good friends, thus spending most of their free time in the center together.
>
> In addition to the normal rehabilitation activities, the center strives to establish strong social engagement and community building activities amongst its residents. Hence, the center is rich with various social facilities accessible to its residents, including shared physical spaces (for example, comfort zones, gardens, and tea rooms), cafeteria, small theater hall (where live plays and recent films are shown), sport halls, etc. Those facilities aim to support the healing and rehabilitation processes, by promoting self-awareness and independent living. Various popular NUI technologies, for instance touch surfaces and motion detection technologies, are deployed in most of the center areas. Those technologies aim to enhance the bodily experience within various center's facilities, ranging from normal home appliances (for example, TV sets) to public devices (for example, vending and ticketing terminals).

The scenario is further labeled with 3 situations to increase the readability and enhance the referencing.

> Aimee and John have just decided to buy two theater tickets for two different theater evening shows from the ticketing terminal available in the theater hall. Consequently, they head to the theater hall to buy the tickets in advance, in order to avoid the peak time in the evening.

Situation 1

> Aimee takes the initiative to interact with the ticketing terminal first. By default, the terminal offers interactiv-

ity based on either direct touch or a basic set of motion gestures. This basic gesture set consists of two simple hand swiping gestures for forward-backward navigation and a touch-in-air gesture for item selection. As Aimee comes in close vicinity with the terminal, her physical abilities and preferences, which are set in advance based on her dedicated rehabilitation plan, are identified. Her current interaction preferences are set to engage head and leg movements. The main purpose is to train and challenge her balance activities as part of her daily rehabilitation routine. In Amiee's case, the terminal's default interactions are clearly not adequate. Accordingly, the terminal reconfigures interaction possibilities to best match this context and an ensemble of interaction alternatives based on head and leg movements is suggested.

The suggested interaction ensemble substitutes the hand swiping gestures with right heel rotation gestures (i.e., clockwise and counterclockwise rotation gestures). It also substitutes the selection gesture with a simple head rotation movement (i.e., vertical head rotation movement). As soon as the ensemble is put into place, Aimee starts navigating a simple menu with left and right foot rotation gestures. Both, the show and the seat are selected by simply gesturing with her head.

Situation 2

Likewise, John steps forward to the same ticketing terminal to buy his own ticket. The terminal recognizes his abilities, accordingly suggests to replace the single hand swiping gestures with two arm swiping gestures (i.e., left arm swipe to the left and right arm swipe to the right). Moreover, replacing the selection task by raising a hand to the front gesture. John successfully issues his ticket by performing the required interaction gestures.

After successfully buying the tickets, John and Aimee head back for launch and prepare themselves for the next rehabilitation session directly after the break.

Situation 3

After the rehabilitation session, John arrives at his room and decides to browse some of his favorite TV channels. Although his hand condition has had improved during the day, he is still not able to handle the TV's motion magic controller correctly. Hence, he switches interactions to his mobile phone (equipped with an enhanced interaction plugin, specifically calibrated to the TV in his room). With aid of his phone, he is now able to provide

> the same set of swiping gestures with a higher accuracy. Almost half an hour later, he leaves to meet Aimee for dinner and then attend the scheduled show at the theater.

This presented scenario is based on a number of assumptions: (1) All of the interaction devices are ensemble-enabled, implying that they expose their identity and capabilities to the ambient space; (2) All applications are also ensemble-enabled, hence announce and subscribe to the required interaction primitives. Additional to the pre-configured default interaction modalities, applications allow also for open interaction streams from external interaction resources; and (3) At least one ambient interaction engine is available, which provides a dynamic creation of ensembles, deployment, and interaction delivery to applications.

As shown in the scenario, different situations require different interaction settings, although using the same devices. Tailored interaction ensembles are created at runtime, based on the users physical profiles and abilities, to enable the user to interact with challenging interactions.

3.6. Interaction Ensembles cases

Figure 3.2 illustrates six different useful cases for interaction ensembles in ambient spaces. Each case is supported with a simplified example for clarity. The examples are correlated to the previously presented "smart rehabilitation center" scenario in section 3.5.

Furthermore, the examples assume three main elements: an interactive application (requires and consumes user input to provide a certain service back to the user), interaction plugins (deliver interaction primitives to applications), and interaction providers (ensemble-enabled interaction devices that acquire elementary interaction tasks).

1. **Full-similar substitution**: This case implies replacing an IP with another IP with the same set of interaction primitives and input tasks. This is useful when a better implementation or a specially tailored IP to a particular disability or situation is available.

   The example is part of situation (3) presented in Section 3.5

   Example: *An interactive TV application requires two interaction primitives for its navigation tasks from a default plugin "A", running on its native magic remote controller. Plugin "A" consumes the horizontal left and right movements of the right hand as an input task. Accordingly, it delivers two swiping interaction primitives (i.e., swiping left and swiping right) to fulfill the application requirements. In the case of a particular hand movement disorder e.g., rigidity, plugin "A" is excluded from the interaction choices. Alternatively, an ensemble-enabled system*

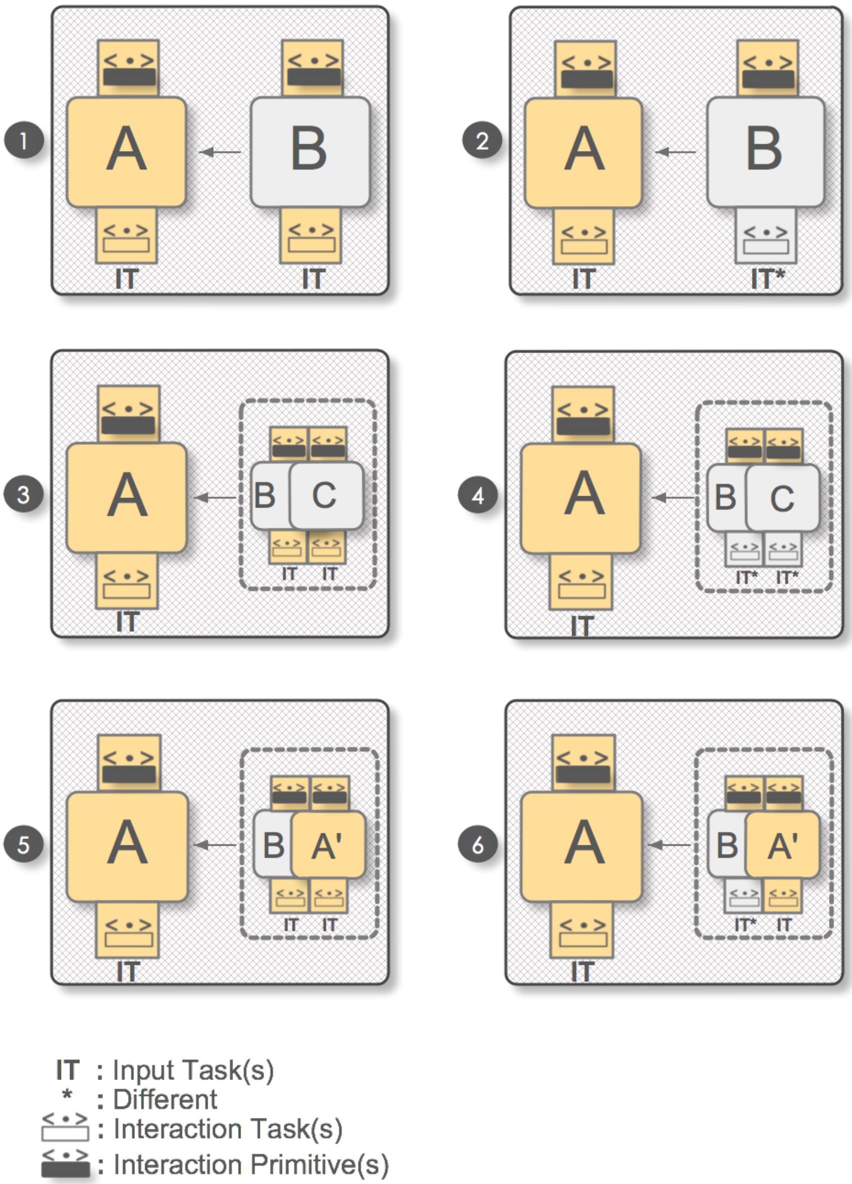


Figure 3.2.: Six useful cases for interaction ensembles: (1) Full-similar substitution, (2) Full-different substitution, (3) Full-similar re-composition, (4) Full-different re-composition, (5) Partial-similar re-composition, and (6) Partial-different re-composition. The IP on the left side of each case (distinctively shaded with a colored shading) presents the original IP. The right side presents either an IP or an ensemble of IPs (color shaded parts indicate similarities to the original IP)

searches for plugins that enable the user to interact and deliver adequate interaction primitives to the application. Accordingly, plugin "B", running on the user's handheld device, is identified and found to be more resistant to jittering hand movements. Plugin "B" consumes the same input tasks and delivers the same interaction primitives but it has a far better optimized implementation for hand rigidity. The substitution of plugin "A" with "B" is considered as a full-similar substitution.

2. **Full-different substitution**: This case implies replacing an IP with another IP with the same set of interaction primitives but different input tasks. This case occurs when a plugin, of different input nature, provides the same semantic output required by the application. This case is often used to choose a plugin that is either more accessible to the user or less affected by the user's impairment.

   The example is part of situation (1) presented in Section 3.5

   Example: *Consider a plugin "A" that consumes the horizontal left and right movements of the right hand as an input task and accordingly delivers two swiping interaction primitives (i.e., left and right swiping) to an application. However, plugin "B" is also available, which consumes the heel clockwise and counterclockwise rotation as an input task and delivers two identical swiping interaction primitives as in plugin "A". The substitution of plugins "A" with "B" is considered as a full-different substitution.*

3. **Full-similar re-composition**: This case implies replacing an IP with composite set of interaction primitives with the same input tasks from multiple IPs.

   The example is part of situation (2) presented in Section 3.5

   Example: *Consider a plugin "A" that consumes the horizontal left and right movements of the right hand as an input task and accordingly delivers two swiping interaction primitives (i.e., left and right swiping) to an application. Due to a hand injury, hand movements are not possible. However, plugin "B" and "C" are available. Plugin "B" consumes the left arm horizontal movement as an input task and delivers swipe left as an interaction primitive. Additionally, Plugin "C" consumes the right arm horizontal movement as an input task and delivers swipe right as an interaction primitive. The substitution of plugin "A" by an ensemble of plugin "B" and "C" is considered as a full-similar re-composition.*

4. **Full-different re-composition**: This case implies replacing an IP with a composite set of interaction primitives with different input tasks from multiple IPs.

   The example is another variation of situation (1) presented in Section 3.5

   Example: *Consider a plugin "A" that consumes the horizontal left and right movements of the hand as an input task and accordingly delivers two swiping interaction primitives (i.e., left and right swiping) to an* *application. Due to a rehabilitation plan, preferences are set to extensively engage head and leg movements as much as possible. Accordingly, plugins "B" and "C" are found available and satisfy these preferences. Plugin "B" consumes the rotation counter-clockwise of the left heel as input task and delivers swipe left as an interaction primitive. Additionally, plugin "C" consumes the right heel clockwise rotation as input task and delivers swipe right as an interaction primitive. The substitution of plugin "A" by an ensemble of plugin "B" and "C" is considered as a full-different re-composition.*

5. **Partial-similar re-composition**: This case implies substituting a partial set of input tasks with other IPs with the same input tasks. This can be illustrated when for example only limited sets of input tasks are utilized by other IPs.

   Example: *Consider a plugin "A" that consumes the clockwise and counter-clockwise wrist rotations as an input task and accordingly delivers two swiping interaction primitives (i.e., left and right swiping) to an application. Due to a hand injury, a limited range of motion is imposed, hence counter-clock rotations are no longer possible. However plugin "B" is available. Plugin "B" consumes the heel counterclockwise rotation of the left leg to generate a swipe left gesture as an interaction primitive. The substitution of swiping left interaction primitive in plugin "A" by the same primitive from "B" is considered as a partial-similar re-composition.*

   In Figure 3.2, "A'" is used to indicate a subset of "A"

6. **Partial-different re-composition**: This case implies substituting a partial set of input tasks from other IPs but with different input tasks. This can be illustrated when, for example, only limited sets of input tasks are utilized by other IPs.

   Example: *Consider a plugin "A" that consumes the clockwise and counter-clockwise wrist rotations as an input task and accordingly delivers two swiping interaction primitives (i.e., left and right swiping) to an application. Due to a hand injury, a limited range of motion is imposed, hence counter-clock rotations are no longer possible. However plugin "B" is available. Plugin "B" consumes the left movements of the front left hand to generate a swipe left gesture as an interaction primitive. The substitution of swiping left interaction primitive in plugin "A" by the same primitive from "B" is considered as a partial re-composition.*

Various research publications in the literature provide evidence that rehabilitation and training processes can be supported and improved by normal interaction with interactive devices. Examples from using mobile devices include SensorShoe [90] and MusicWalk [188] for gait training and analysis of parkinson patients using mobile phones respectively. As seen in the different ensemble cases, NUI ensembles' interoperability features a number of useful scenarios such as replacing dominant hand with non-dominant hand interactions, force with position grip interactions, and hand with feet interactions. Ensembles may enforce a better performance and integration of users within their known physical abilities and may also be increasingly useful in physical therapy and rehabilitation, e.g., maintaining and improving mobility, flexibility, strength, gait speed, and quality of life.

More about interface challenges in ambient spaces are presented in sections 1.2 and 2.2.6

Moreover, the discussed scenario attends to the interaction challenge in ambient spaces as indicated by [137]; namely heterogeneity and distributivity (i.e., expecting a variety of interaction resources and possibilities available to users), dynamic media mobility (i.e., the availability of interaction possibilities is dynamic and changing), and user mobility (i.e., dynamic and changing user needs due to high mobility).

3.7. Building and orchestrating Interaction Ensembles

This section discusses the main conceptual building blocks for ensemble-enabled interactive systems. In Figure 3.3(a), both interaction devices (interaction providers) and applications (interaction consumers) are based on arbitrary technical platforms, built by interaction designers and by application developers respectively. The communication may take place in various forms but must nonetheless aim at a high level of interoperability and compatibility. An ensemble-based systems should prevail communication problems by avoiding bandwidth intense payloads, e.g., images. It uses highly optimized interaction tasks data types based on an extended list of primitives including position, movement, rotation, etc., and optimized interaction primitive data types for the consuming applications such as selection, panning, etc. As indicated in Figure 4(b), interaction designers create their IPs and publish them using infrastructure's interaction publishing front end. Moreover, interactions are provided by a single atomic IP or by an orchestrated ensemble of IPs. Interactions provided by an IP may well have a number of implementation alternatives if multiple interaction resource provides alternative implementations of the same IP as shown in section 3.6.

In following subsections, this conceptual view is elaborated and the main design choices are discussed: (1) Interaction Plugins conceptual design; (2) Anthropometric driven matching and presentation of Interaction Plugins; (3) On-demand wiring of interaction resources; and, (4) Community-based designing and sharing of NI.

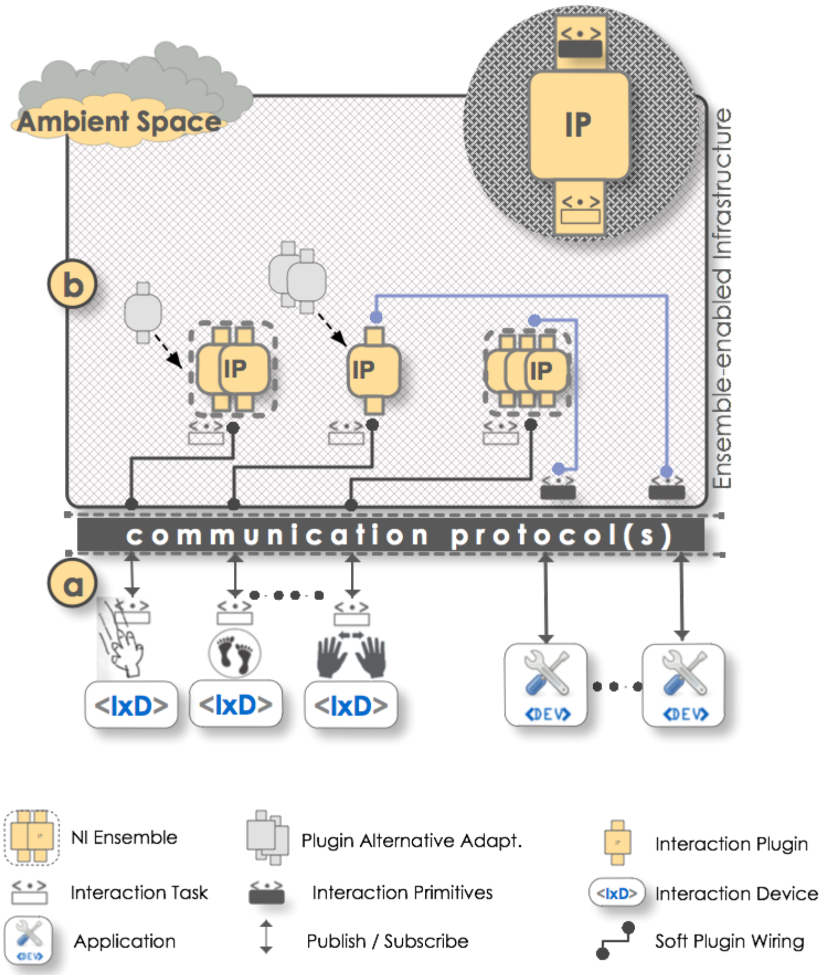


Figure 3.3.: Ensemble-enabled system conceptual diagram

3.7.1. Building Interaction Plugins

Interaction Plugins are the core building blocks for interaction ensembles. As already defined, IPs are standalone deployable and executable units. The main purpose of those units is to convert the input tasks into interactive primitives that can be consumed by applications.

The Interaction Plugin concept is defined in section 3.4

In section 1.2, we have covered some of the interface challenges in ambient spaces. The challenges of user interface adaptation and adoption in ambient spaces are mainly caused by the emerging issues such as heterogeneity and distributivity, dynamic media mobility, and user mobility [137]. In addition, we have argued in section 1.4 that kinetic interactions, the focus of our research, are currently hindered due to the lack of systematic consideration of the increasing diversity of user populations in ambient spaces and effective means for documenting, adapting and deploying interaction techniques. The concept of IP, as a novel approach for sharing kinetic-based NI techniques in ambient spaces, is proposed to address these hindrances. To the best of our knowledge, there is no research specifically targeted at community-based creation and sharing of encapsulated NI techniques. Hence, we have defined our own structured approach based on three main design characteristics:

- Matching users and NI physical context;
- Precise and extensible NI descriptions (human and machine readable); and,
- Flexible deployment of NI plugins at runtime.

Ability and movement profiles are discussed in section 3.7.2

As seen in Figure 3.4, IPs reside in a plugin repository, which is accessible to interactive devices. Plugins should be designed to be dynamically deployable at runtime (as discussed in subsection 3.7.3). Plugins are associated with three different profiles to enable the required IP filtering and matching operations. Those profiles will be elaborated in the following section.

3.7.2. Anthropometric driven description and matching of Interaction Plugins

The disability challenges are discussed in Section 2.2.5 and the implications of aging on AAL are covered in 2.1.2. As NUIs are inherently affected by a wide range of physical impairments and disabilities, we opted for an anthropometric approach for describing, discovering and presenting IPs. Regardless the internal working of an IP, plugins should provide sufficient information about the physical movements involved in the interaction. Such information enables plugins to reason about all physical movements required for a given interaction scenario. Moreover, plugin discovery mechanisms utilize this information to optimize the selection of suitable IPs for a given scenario and user.

The conceptual design revealed three essential building blocks (information components) for anthropometric driven IP:

- **Movement Profile**: First, movement information is utilized by a movement component represented by a profile. Movement profile is used to describe the essential physical movement required by the interaction from the user (from an interaction capture point of view). For instance, for a balance interaction technique, this profile precisely describes the body balance, position, and posture required by the interaction. This profile should be intrinsically flexible in oder to capture various abstraction levels for describing movements. Interaction developers should

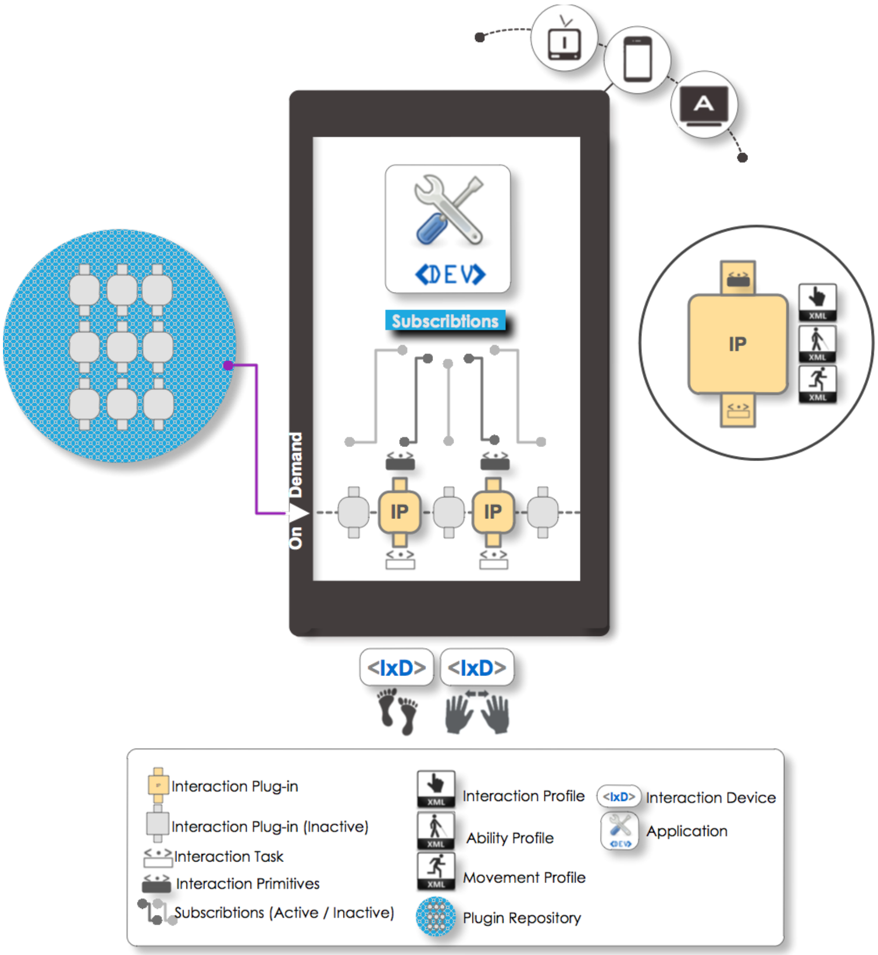


Figure 3.4.: A conceptual overview over Interaction Plugins deployment

be able to set the adequate abstraction details of the movement, to reflect the essential and important aspects of the technique. Interaction developers may also describe the detailed micro-aspects of the movement, which may lead to a complex but precise movement description. In contrast, developers may describe the movement on a macro level while leaving a lot of movement details out. The detailed design and implementation of this profile is covered in section 6.4

- **Ability Profile**: Second, the ability profile defines the physical abilities that are required for an appropriate execution of NI. It is composed from two building blocks: major physical activities vital for the interaction, such as holding, standing, balancing, lifting, walking, etc; and physical disabilities that may impact the quality execution of the interaction. Both aspects are evaluated through an impact score for each, which defines their importance and impact to the quality of interaction execution. We argue that movement and ability profiles are essential for designing interactions "for all" instead of focusing on a limited population percentile and to avoid inaccessible interfaces. The detailed design and implementation of this profile is covered in section 6.5.

- **Interaction Profile**: The third component in our model is the interaction profile that includes the main interaction semantics, including the interaction primitives such as pointing, selecting, dragging, etc. This is important in ambient spaces because it provides an indication of the main use of the interaction technique and its offerings. This profile is also used by the applications when subscribing to available interaction plugins that provide the same type of interaction primitives. More about design and implementation aspects of interaction profile is covered section 4.5.3.

3.7.3. On-demand wiring of interaction resources

The implementation aspects of IPs deployment are extensively covered in Chapter 4

As we have covered in section 2.2.6, despite the novel research contributions in this area in the two decades, the fusion of NI techniques into ensembles of interaction techniques is still a rather unexplored area. The combination of hand and foot input for example has gained only little attention according to Daiber et al. [42].

Principally, distributed and modular device interfaces, enabled by accumulative hardware ensembles, proved useful for interactions where additional interface elements can be added as needed to devices, like a larger screen, a mouse, speakers, a keyboard, a projector, an e-ink screen, and so on. Nevertheless, to the best of our knowledge, interaction techniques for NUI have never been investigated similarly, even though ambient spaces require highly adaptive and modular approaches due to their highly demanding context requirements. Much of the literature focuses on using a limited part of the body. Interactive systems that incorporate the gross motor skills and utilize the kinesthetic sense have not been thoroughly investigated despite the growing number of implementation examples [56]. Nonetheless, there is a strong emerging motivation to explore new potential in designing for the whole body in motion as shown in section 2.2.3. Against this background, users are expected to interact with multiple interaction techniques simultaneously employing multiple body parts and different motor skills. Hence, interactions in NUIs are expected to not only play individually but also to play as part of an ensemble. Moreover, interaction techniques are expected to join and leave the ambient spaces at anytime due to the user mobility and due to the changing availability of interaction resources.

Mechanisms for interaction for dynamic interaction deployment at run time are becoming essential needs for future ambient spaces. Interaction sharing can only be realized with a comprehensive approach for on-demand discovery, selection, and integration of interaction modules at runtime. Support for runtime provisioning of suitable IPs should be based on the user's capabilities and a plugin's required physical movements. Figure 3.4 illustrates a conceptual view of IPs deployment on mobile devices. IPs are discovered, loaded into devices on demand, activated seamlessly, and exposed to interactive applications through an easy-to-use subscription mechanism.

IPs are hosted by a plugin repository, where they can be requested and downloaded on-demand by applications. IPs expose their offered interaction capabilities through interaction primitives (e.g., pointing or selection) through which interactive applications can subscribe to the plugin. After deployment, the IP resides in the user's ensemble-facilitator device (i.e., interaction hub) in an inactive state. This state is changed to active once a subscription is received by the plugin, in order to allow plugins to only occupy the needed interaction and computation resources. This allows to attend to certain amount of plugins at a time, leading to save resources and computation power on the device. During runtime, interaction events are fired and delivered to the application. In our current model, applications can be informed about which plugin to subscribe to and activate based on the physical abilities and qualities required by an application in a particular user context. In section 4.5, the actual use and illustration of those profiles by ensemble-enabled systems is discussed in details.

3.7.4. Community-based designing and sharing of NUI

Community-based designing and sharing of NUI are very important in order to easy the use of NUI in application development, enhancing application adaptability, and promoting wide deployment of NI based applications. To our knowledge, there is no research specifically targeted at community-based creation and sharing for natural interaction techniques (as Interaction Plugins). This puts this outlined work forward to be the first framework to study this concept rigorously, as it aims at community-based description of NUI techniques and contexts, supporting both ambient interactive system designers and application developers.

A successful community-based designing and sharing of NUI requires various important design elements. Those elements are of equal importance to users, designers, and interaction recognition systems:

- Standard documentation and dissemination strategies for interaction in NUIs (discussed in Chapter 4 and Chapter 6);

- Interaction authoring and documentation tools (discussed in section 6.2);
- Deep understanding of designers' documentation practices (discussed in Chapter 5);
- Deep understanding of users' learning practices (discussed in Chapter 5);
- Interaction guidance and demonstration tools (out of the scope of this thesis); and,
- Seamless ad-hoc and runtime interaction deployment mechanism (discussed in Chapter 4).

3.8. Outlook

In this chapter, the vision and the theoretical approach proposed in this dissertation are presented. The chapter starts by arguing that designing interactions for NUI devices is challenging, because of the limited support for a full exploration of the potential of motor interaction, even when optimized for the considered impairments. Hence, often mismatch problems between users' needs and devices' offers exist. Interaction Ensembles are proposed as an alternative approach aims at utilizing dynamic compositions of interactions in NUIs. The approach fosters soft-wired applications and devices in order to overcome the limitations of the static binding and to address one of the most challenging requirements in pervasive environments, namely the "come as you are" requirements.

The conceptual design of Interaction Ensembles notion is presented, which form the ground for the dissertation investigation. The conceptual design is tackled on four different sides including IPs design, anthropometric driven matching and presentation of IP, on-demand wiring of interaction resources, and community-based designing and sharing of NI.

In the context of this dissertation, various research activities are carried out towards those design issues. In Chapter 4, we build further on this conceptual design and we focus present a technical design and implementation of the STAGE framework. Chapters 4 presets a detailed investigation about NUI documentation and dissemination. Tools and languages for documenting and sharing interactions are discussed in section 6.2. This chapter also includes designing and implementing a tool for documenting and authoring interaction called Interaction Editor. Moreover, community NUI documentation practices and users' learning habits are discussed in Chapter 5.

4

STAGE: Ensemble-enabled Interactive Framework

4.1. Introduction

Parts of this chapter appear in [5]

In the previous chapter (especially in section 3.7), a conceptual model for designing and deploying interaction techniques was presented. In this chapter, we tackle the issue of interaction deployment in ambient spaces. The fundamentals of the technical implementation and realization of the Interaction Plugin concept towards interaction deployment in ambient spaces are presented and discussed in details.

The presented implementation in this chapter is one of many possible ways to realize the proposed concept. Hence, the proposed implementation should not be necessarily considered as an ultimate implementation solution, instead the implementation aims to demonstrate and evaluate the feasibility of the approach. The current implementation utilizes the wide adoption of mobile devices for rich personalization, customization, and context acquisition in ambient spaces. Hence, this work fosters the use of mobile devices as customized and personalized interaction hubs.

This chapter is organized in eight sections. First, the chapter starts by presenting a review on dynamic component integration and adaptation, focused on the ambient computing research (4.1.1). Furthermore, the chapter discusses context modeling and deployment based on the Dynamix framework (4.1.2). Second, the implementation architecture for the STAGE framework is presented in section 4.2. Third, the implementation of IP is presented in section 4.3. Fourth, provisioning IPs is discussed in section 4.4. Fifth, section 4.5 presents the main building blocks for STAGE-enabled applications including the presentation of the Interaction Manager; Ability Manager; Interaction Profile Manifest; and, the application's interaction sequence. Sixth, section 4.6 presents our performance evaluation including the experimental setup and procedure, as well as the results. Seventh, IP showcases and examples are presented in section 4.7.

4.1.1. Dynamic component integration and adaptation in ambient computing research

Modeling, designing, and managing context information are significant problems facing application developers, particularly in the area of mobile computing. In [44] and [175], researchers pointed out that those problems are mainly caused by the absence of appropriate support for rich context types, approach extensibility, and easy-to-use context frameworks.

In the literature, a variety of approaches have been proposed for solving those problems, through specialized middleware [142], context servers [50], and environmental instrumentation [136]. However, such approaches fail to scale in many application scenarios, as in wide-area mobile scenarios [44] [22].

Recently, an increasing availability of commodity sensors offers a wide range of sensors data such as orientation information; acceleration information; geo-location; proximity; light levels; camera and microphone streams; etc [124]. In response, we are seeing an increasing interest in mobile context frameworks capable of providing abstractions for sensing; context modeling and representation; service discovery and binding; etc [33]. Such information is very useful to mobile applications to fluidly adapt to the user's changing environment and the context of use [124].

Dynamic component integration has been a rich and yet challenging investigation aspect for the ubiquitous and pervasive computing research. The dynamic component integration approach enables software components to be discovered, downloaded, and integrated on-demand, as a means of adapting an application's behaviour and enhancing its features [152]. Recently, such investigation is successfully applied to mobile environments as an interesting major milestone in this application domain. One of the first projects to support dynamic component integration on Android using Open Services Gateway initiative (OSGi) containers was the Mobile USers In Ubiquitous Computing Environments (MUSIC) system [37]. The Context-Aware Machine learning Framework for Android (CAMF) promotes plugin-based adaptation on Android. The framework introduced processing widgets as discrete abstractions used to hide machine learning complexities [174]. Furthermore, the Funf Open Sensing Framework[1] promoted statically-linked context modeling plugins integration.

More recently, the Dynamix framework [33] was introduced as an open plug-and-play context framework for Android. It supports automatic discovery and integration of plugins for context acquisition (sensors) and modification (actuators) at runtime; 3rd-party context plugin support; and custom context representations based on Plain Old Java Objects (POJOs),

[1] http://funf.org, accessed on 25.03.2014.

which enables the creation of arbitrarily complex context event objects that are easy to parse and use.

Investigating the available mobile-based platforms for context modeling and dynamic deployment revealed that the Dynamix framework is the most adequate for our implementation, as discussed in detail in the rest of this chapter.

4.1.2. Context modeling and deployment with the Dynamix framework

The Ambient Dynamix is a plug-and-play context framework for Android. The framework was first introduced by Carlson and Schrader [33]. It was proposed as a rich framework for on-demand discovery and runtime integration of context sensing and acting capabilities in wide-area mobile contexts. Interestingly, Dynamix features strong and flexible discovery and deployment of context plugins at runtime. The framework was successfully used to model and deploy context in different none-HCI related scenarios as in [31][32][30]. The two features are essential building blocks for the implementation of the Interaction Ensemble concept. In addition to its capabilities, the framework is publicly available as an open source project. To the best of our knowledge, our work is the first to utilize and build on this framework for deploying interactions in ambient spaces.

Due to its relevance to our current implementation, we will describe the internal components of the framework based on [33]. The full specifications and more extensive discussion can be found in the same research paper and in the Dynamix online portal[2].

Dynamix runs as a background service on Android-based devices. This service allows multiple applications to subscribe to Dynamix, in order to receive context events accordingly. Upon subscription, the Dynamix-enabled applications receive context events from the available plugins. Principally, a Dynamix-enabled application is a normal Android application that implement the necessary Dynamix Application Programming Interfaces (APIs).

Figure 4.1 illustrates, in details, the main components of the framework. In the bottom side of the figure, various context resources deliver various raw sensor data to the framework. The data will be then modeled by Dynamix according to the available context plugins. A context plugin is a standalone deployable OSGi container. This container allows for seamless runtime deployment (i.e., installation, de-installation) according to the application needs.

[2]http://ambientdynamix.org, latest access on 35.03.2014.

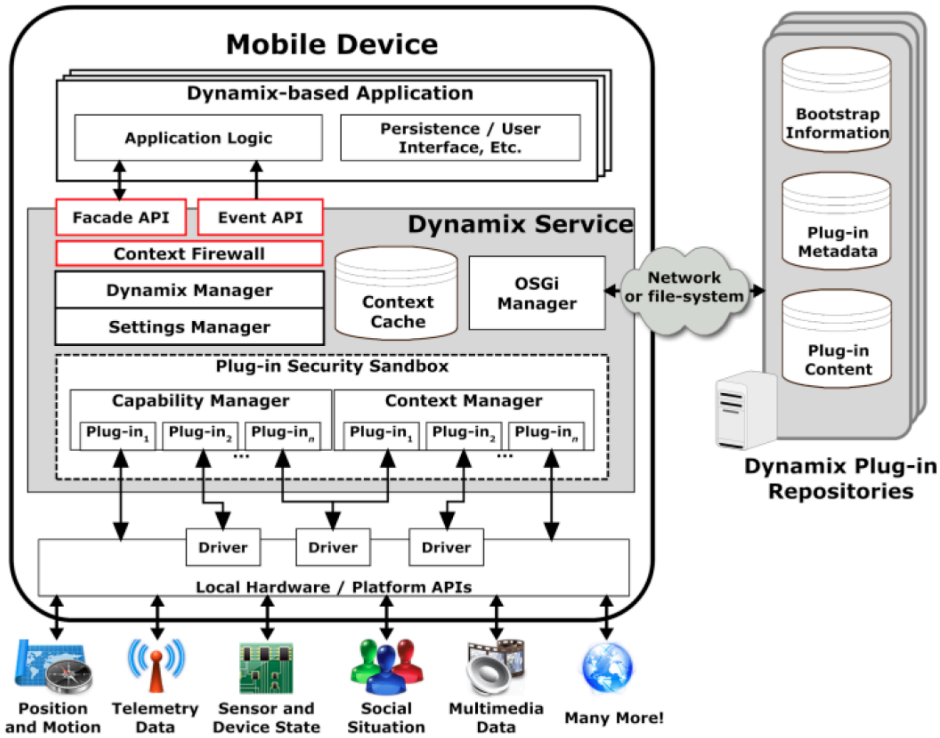


Figure 4.1.: Overview of the Dynamix framework (from [33], used with permissions from the author.)

The Plugin Security Sandbox is designed to allow the framework to securely control and manage every plugin individually, without affecting the rest of installed plugins. The Sandbox provides the required insolation mechanisms to control the flow of context events to applications. At the same time, it allows Dynamix to take over and terminate misbehaving plugins seamlessly, without affecting the rest of running processes.

Dynamix handles security aspects using a dedicated component called the Context Firewall. This firewall grants the user a full control over the available plugins. Likewise, it enables the user to regulate the applications' accessibility and use of context data. Through a manual configuration interface, the user is able to set different levels of privacy constrains on each plugin individually, according to the privacy risks posed by the plugin. The user is able to adjust those security levels flexibly at anytime according to their needs and preferences.

In order to reduce the loading on context modeling, the framework allows context plugins to catch modeled context information for a dedicated validity duration. This process is controlled and managed by the Context Cache unit.

The OSGi Manager resides at the core of Dynamix implementation. It is mainly dedicated for all plugins deployment actions. It's most essential actions are the installation and un-installation of plugins from their OSGi containers [33].The OSGi framework allows for seamless integration of software units (called bundles) at runtime. Internally, the OSGi framework is composed of multiple layers responsible for bundle execution, bundle management, bundle life cycle, service and binding management, and security control. Dynamix is build based on the Apache Felix OSGi [33], which allows bundles deployment on native Android-based devices. Moreover, the Apache Felix OSGi preserves most of the native Android capabilities, most importantly background services, multitasking, and inter-processes communication. All of those features are utilized intensively by Dynamix.

Currently, OSGi is one of the most adopted solutions for software integration for Java [33].

There are various communication aspects in Dynamix. We will focus, herein, only on the communication between plugins and applications. This communication is the most relevant in the context of our development. In its current implementation, Dynamix supports POJO objects to encode the events shared between plugins and applications. The use of POJO objects allows both sides to work with Java objects, hence increasing the operability and reducing the programming and modeling load.

The communication between the Dynamix framework and Dynamix-enabled application is featured using the Facade and Event APIs. The Facade APIs controls all requests for context modeling support. For instance, the applications subscriptions to a particular context type. The Event APIs controls the context events delivery to subscribed applications. For instance, delivering a new context event to an application.

Dynamix offers an extensive feature list that exceeds the deployment features to the Dynamix power profiling and beyond. Nonetheless, those features will not be discussed here, as they are out of the focal interest of our implementation.

4.2. STAGE architecture and implementation

In our implementation, we leverage Dynamix as a mechanism for sharing natural interactions on mobile devices, due to its unique capabilities and flexibility, especially related to dynamic discovery and deployment of suitable context plugins during runtime. Hence, it is feasible to adopt and extend context plugins for interactions. STAGE considers mobile devices as personal interaction hubs in ambient spaces, which facilitate interactions to any ambient object or service available in the local vicinity of the user, which has an existing plugin. External ambient applications (none-standard Dynamix applications) may still benefit from the complete set of services STAGE and Dynamix provide by implementing an own background service (i.e., adapter),

The theoretical concepts for building ensemble-enabled systems are covered in section 3.7 which enable the application to consume context and interaction events. In our walkthrough scenario presented in section 3.5 (situation 3), the user's phone provides all interaction services to a smart TV. In this scenario, the phone acts as an interaction hub to facilitate interaction ensembles and interaction modeling on behalf of the TV.

Our current implementation is based on Dynamix release version - 0.9.58

Figure 4.2 illustrates our underlying technical approach for implementing NIs as deployable and shareable IPs, based on the Dynamix framework on the Android platform. As intended, the Dynamix framework runs as a background service (Dynamix Service) and is situated between Dynamix enabled applications and the device's interaction resources (i.e., interaction devices).

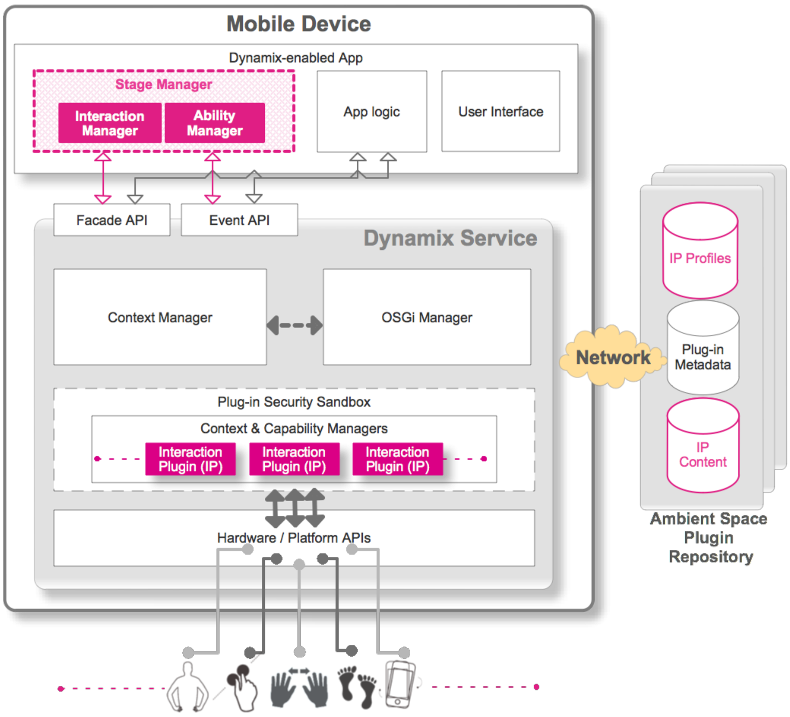


Figure 4.2.: STAGE realization based on Dynamix

To avoid a reductionist infrastructure design (considering technical metrics only), both interaction and technically orientated metrics should be considered equally, as pointed out by Edwards et al. [48]. In addition to various technical metrics such as throughput, latency, and scalability, more human-centered interaction metrics should be also considered such as installability, evolvability, predictability, and intelligibility. Those metrics impose unresting challenges in ambient spaces including: managing on-device resources (low-level interaction capture and preprocessing), eventing and networking issues, addressing extensibility and modularity needs, and user experience.

The STAGE technical implementation appears mainly in two areas: first, the interaction application side; second, the context plugin side. Dynamix-enabled applications are standard Android applications with extra context modeling functionality provided by a local Dynamix Service. In STAGE implementation, two new components are introduced, namely the Interaction Manager and Ability Manager, to adjust the Dynamix framework to our needs. Additionally, Dynamix context plugins were adopted and extended in the IP implementation as discussed in section 4.3.

Interaction Manager is presented in section 4.5.1 and the Ability Manager is presented in section 4.5.2.

4.3. Implementing IPs as deployable interaction units

As already briefly discussed in this chapter, an IP is designed and implemented as a Dynamix context plugin. From the technical implementation point of view, an IP is essentially an OSGi Bundle containing the plugin's logic and context acquisition code. In our case, Dynamix is responsible for handling the plugin's lifecycle, based on its embedded OSGi framework.

Interaction Plugin conceptual design is covered in section 3.7.1. Full reference for the Dynamix plugin development reference can be found in the Ambient Dynamix portal.

Figure 4.3 illustrates the technical implementation of the IP. The distinctly tinted components in the diagram indicate the core STAGE components and extensions.

- **Context Acquisitions and Modeling**: This module resembles the logic required to access the context provider and its raw context data. It is important to recognize that the raw context data resources may be local resources (i.e., running on the same device, for example a built-in orientation sensor) or remote sources (i.e., accessible ambient context providers available in an accessible ambient space, for example an external camera sensor).

- **STAGE Handler**: This module is responsible for modeling the interaction events based on the received information from the previous module.

- **Plugin factory**: The Dynamix Plugin Factory provides the required mechanisms for plugin instantiation. This part is handled completely by Dynamix.

- **PluginRuntime**: This component contains the plugin's core lifecycle methods. Dynamix allows three different behaviours, which govern the

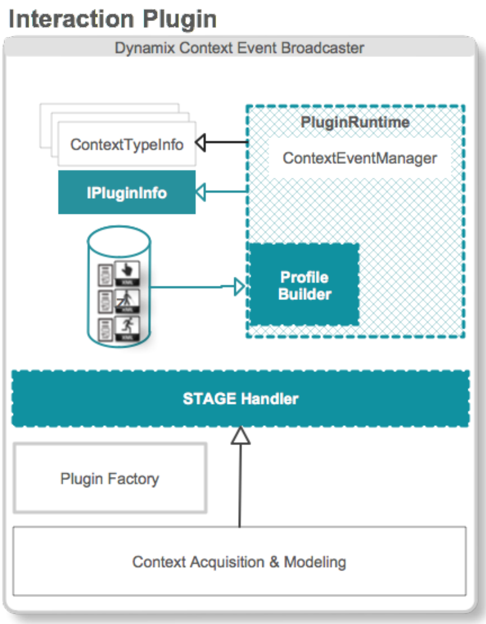


Figure 4.3.: IP internal architecture

plugin. The most relevant for our implementation are the autonomous and reactive types. Autonomous plugins model context information continuously and broadcast events to subscribed applications. Reactive plugins model context information in an explicit response to a particular context request.

- **Profile Builder**: As part of the PluginRuntime component, the profile builder is responsible for serializing and building the interaction, movement, and ability profiles associated with each IP. Those profiles are modeled as XML embedded in the plugin's file structure and accessible though the IPluginInfo object.

- **ContextTypeInfo Objects**: Based on the Dynamix implementation, context plugins may have datatypes presenting the type of the supported context by the plugin. Those datatypes are presented by the ContextTypeInfo objects. In our implementation, those context type objects are used as interaction primitive objects.

- **IPluginInfo Object**: We have introduced a new plugin information datatype called IPluginInfoObject. This datatype is important for any IP, as it contains the necessary movement, ability, and interaction profiles (normally requested by Listing 2).

The IP follows the conventional OSGi bundle's lifecycle that is managed by Dynamix internally. The lifecycle of an OSGi bundle is shown in Figure 4.4. The bundle status is bound to six different states: installed; resolved; starting; active; stopping; and uninstalled. Those states are controlled by five transitions: Install (the life cycle starts once the bundle is stored persistently in the framework); Start (resolving the package dependences and move the bundle into the resolved state; respectively, the bundle gets instantiated and moves to the active state); Update (updating an installed bundle); Stop (stopping the bundle and perform clean up processes); and, Uninstall (uninstalling the bundle).

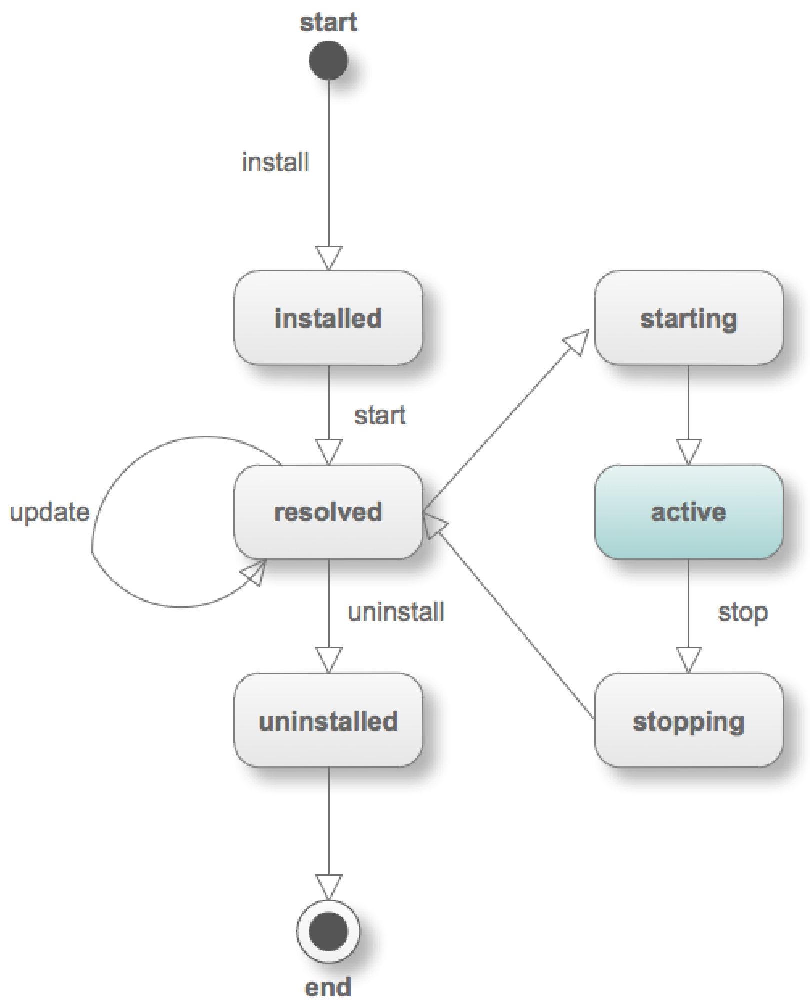


Figure 4.4.: IP lifecycle as an OSGi bundle (adopted from [128])

The Dynamix framework exposes those states partially to the plugin's developer. Additionally, Dynamix introduces new states and transitions as well. Its most important lifecyle states and transitions are: New (upon plugin instantiation using the "init" method), Initialized (after a plugins's successful initialization), Started (after a successful plugin start processes), and finally Destroyed (after calling the "destroy" method).

IP performance evaluation is presented in section 4.6

Dynamix provides mechanisms for context event caching, whereby an event expire period can be specified. Despite the availability of this functionality, we have deduced not to implement this feature for IPs, in order to insure up-to-date responses to the user's actions and to avoid caching problems due to high eventing rates (often needed for user's interactivity).

4.4. Provisioning Interaction Plugins

Provisioning IPs undergoes a number of phases as illustrated in Figure 4.5. In the first phase, the interaction techniques passes the usual design and implementation processes. In the second phase, the interaction developer (i.e., designer, developer, or team) makes sure that the techniques satisfies the function and utility defined in the interaction design. In the third phase, the interaction developer should define the interaction's movement profile, based on an acceptable level of movement description (discussed in chapter 6); define the interaction's ability profile, based on the most important physical qualities that impact the interaction; and assign the interaction semantics, based on the envisioned utility of the technique. In the fourth phase, the interaction developer wraps the interaction's internal logic as a Dynamix plugin. In the fifth phase, the IP is published to an accessible repository. Dynamix supports two types of accessible plugin storage locations (file-system or network). Our implementation supports the later type as shown in Figure 4.2. The repository hosts the plugins OSGi bundles and the corresponding Context Plugin Description (CPD) XML documents.

The decision to publish the designed IPs to private or public repositories is left to the developer of the plugin. In many cases, this decision is influenced by the application scenario and supported users. Setting up plugin repository is made possible with a reduced server-side complexity. In fact, the setup is simplified to an accessible file system with a Dynamix's CPD document and a bundle for every IP. This requires minimum management and infrastructure support on the server side. All deployment and repository management functions are controlled by the Dynamix Management application, which offers users a full control over trusted plugin repositories and deployment preferences.

By successful performing those phases, interactive application developers can easily create Android-based applications on the Dynamix framework and available IPs (according to the required semantics). IPs are also compliant with the AmbientWeb extension of Dynamix, which exposes Dynamix's

Figure 4.5.: IP provisioning lifecycle

plug-and-play context capabilities to Web applications running in unmodified mobile Web browsers [30]. Nonetheless, a full support for the AmbientWeb applications is envisioned in future releases of STAGE.

4.5. Implementing STAGE-enabled applications

Dynamix applications are designed to subscribe to dynamix and receive modeled context events. Those two features ease the implementation of context-based applications dramatically. Nonetheless, Dynamix-enabled applications are not fully adequate to our conceptual design. Hence, STAGE introduces two important extensions to the architecture, in order to facilitate the use of this framework for Natural Interactions, on the application and plugin levels. The changes and extensions are distinctly marked (i.e., colored) in Figure 4.2. The STAGE components are further illustrated in Figure 4.6.

Figure 4.6 shows two essential new differences to the original Dynamix application architecture. First, the STAGE Manager, the core of our extension to the application side, which contains the Interaction and Profile Managers. Second, the Interaction Profile Manifest, which encapsulates all necessary information about the required and relevant interaction capabilities by the application.

4.5.1. STAGE Interaction Manager

The Interaction Manager class is developed to control the activation of the available IPs, based on the ability, movement, and interaction profiles. Both the Interaction Manager and the application logic separately communicate with Dynamix service through its Facade API, which enables apps to request and control context modeling support; and its Event API, which enables apps to receive framework notifications and context events.

The profiles are first introduced in section 3.7.2. The interaction, movement, and ability profiles are elaborated in section 4.5.3, section 6.4, and 6.5 respectively.

Interaction Plugins are tailored OSGi-based Bundles, which are loaded into the Dynamix embedded OSGi container at runtime. Once loaded and activated, the plugin sends interaction encoded events to subscribed applications (as interaction primitive events) using POJOs. An

Internet-based plugin repository hosts IP Bundle files, the IP profiles, and (optionally) additional related plugins.

In addition to usual Dynamix context sensing tasks, the IPs, hosted by Dynamix, can be queried by the application's Interaction Manager to access the information encoded in the IP profiles, ranging from the physical abilities required (e.g., body parts and range of motion) to the interaction semantic (e.g., selection or positioning). Currently, the Dynamix Service provides all plugin discovery services, plugin filtering based on the interaction requirements (interaction primitives required), and plugin installation support. Filtering and activating the available interaction plugins are currently handled by the Interaction Manager; however, we envision a highly configurable plugin selection and activation functionality integrated as a core extension of the Dynamix architecture in future releases.

The STAGE Interaction Manager is composed from the following components, as shown in Figure 4.6:

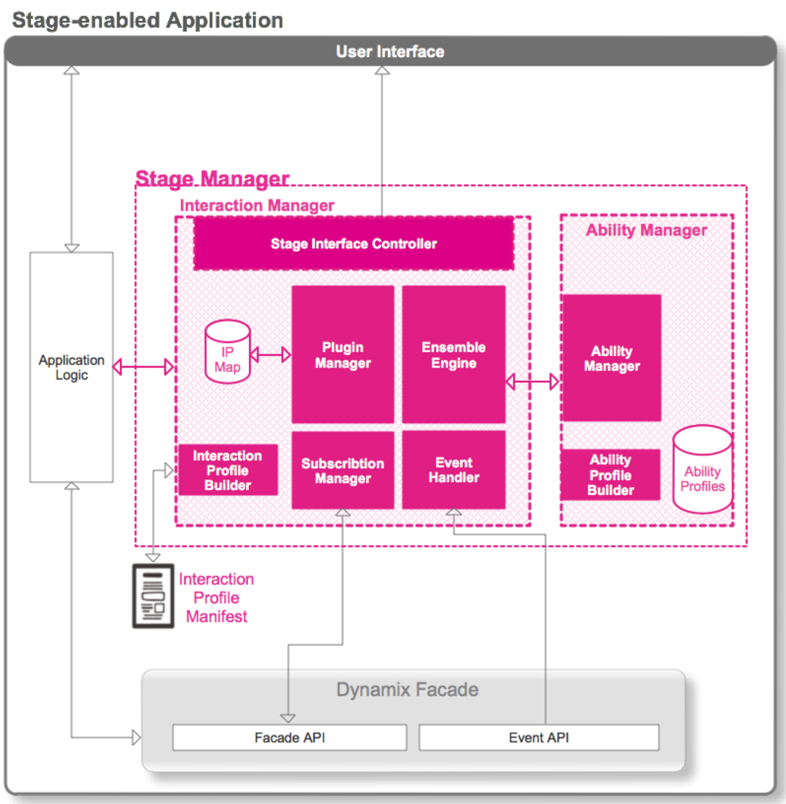


Figure 4.6.: STAGE-enabled application architecture

- **Subscription Manager**: This module is responsible for communicating directly with the Dynamix facade API. The purpose of this module is to control and mange the subscription processes to IPs.

- **Event Handler**: This module consumes all Dynamix context events and filters all relevant Interaction Events. Once filtered, detected Interaction Events are then sent to the Ensemble Engine.

- **Interaction Profile Builder**: This module is essentially an XML parser responsible to extract and serialize the content of the Interaction Profile Manifest into a runtime object. This object can be used by the Plugin Manager and the Ensemble Manager to identify the required interaction tasks by the application.

- **Plugin Manager**: The module keeps track of all accessible IPs. Moreover, it handles the status, subscription, plugin information, and discovery requests of all plugins. The manager also has a map of all IPs for easy access and query by other STAGE components, especially the Ensemble Engine.

- **Ensemble Engine**: This module is responsible to monitor interaction resources (i.e., IPs) and build possible Interaction Ensembles adequate for the given context, based on the application's required interaction capabilities, the available IP, and the user's physical ability profile.

- **STAGE Interface Controller**: STAGE-enabled applications require to control the GUI elements (i.e., components) according to the fired interaction primitives from the IPs. This functionality is facilitated by the native Android GUI control mechanisms, which allow native Android code to simulate GUI actions and events programatically. Hence, once an interaction primitive is fired, the corresponding action on the interface is executed. For example, a selection interaction primitive may be interpreted as a button press on the interface.

4.5.2. Ability Manager

This module is responsible to extract the user's ability and disability qualities, based on the available Ability profile and Movement profile. Currently, both profiles are modeled and represented in tailored XML formats. This component is essentially split into the following modules:

The implementation of Movement and Ability profiles are discussed in section 6.4 and 6.5 respectively.

- **Ability Profile Builder**: This module is essentially an XML parser responsible to extract and serialize the content of the Ability Profile into a runtime object. The physical abilities are also kept in a database for quick access and management by the Ability Manager.

- **Ability Manager**: This module provides all necessary information regarding the required physical abilities for executing the interaction tasks adequately by the user. This information is consumed by the STAGE Interface Manager.

4.5.3. Interaction Profile Manifest

Interaction profile, first discussed as part of our conceptual model in section 3.7.2, captures the main interaction semantics including the interaction primitives. The Interaction Profile Manifest is used to identify the required interaction capabilities for the application. The manifest file is constructed in XML format, and contains essentially the required interaction primitives. Figure 4.7 shows the current XML scheme definition of the manifest and listing 1 resembles a simple profile example for an application requiring a single interaction primitive.

The scheme definition is simplified to listing all required primitives by specifying the type of the primitives and their associate plugin details. The type of any supported interaction primitive is represented by the name space of the primitive. Principally, the name space of an interaction primitive consists of the default identifier of the STAGE system ("de.itm.STAGE.nui"), followed by the primitives identifier ("primitives"), and the unique interaction primitive identifier (e.g., "selection"). This primitive is indicated by the full name space of the interaction primitive (de.itm.STAGE.nui.primitives.selection) in the "type" element. While the actual definition of this type can follow the Niezen's ontology for interaction primitives in smart environments [125] (presented in section 3.3), the actual definition of this type is left open to the interaction designer to insure extensibility, which is done by defining the IP ContextTypeInfo Object (discussed in section 4.3).

The manifest allows application designers to restrict certain interaction primitives to a particular IP. This is done by filling the "pluginId" element with the desired IP ID. By default, this element is left empty to avoid any restrictions. In most cases, imposing such a restriction dramatically limits the range of interaction possibilities, in the case of absence of the desired IP. Although such behaviour is not preferable, it is still useful in some cases where the application designer intends to impose a certain behaviour. For instance, restrict authentication related interactions to trusted plugins.

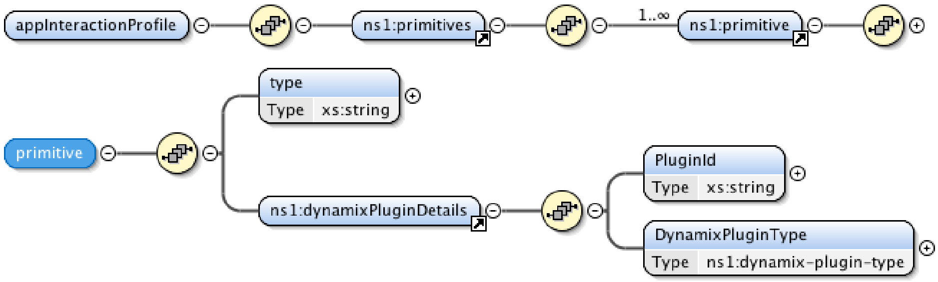


Figure 4.7.: Interaction Profile Manifest XML scheme

```xml
<?xml version="1.0" encoding="UTF-8"?>
<appInteractionProfile>
    <primitives>
        <primitive>
            <type>de.itm.STAGE.nui.primitives.selection</type>
            <dynamixPluginDetails>
                <pluginId></pluginId>
                <dynamixPluginType>AUTONOMOUS</dynamixPluginType>
            </dynamixPluginDetails>
        </primitive>
    </primitives>
</appInteractionProfile>
```

Listing 1: Interaction Profile Manifest sample

4.5.4. STAGE-enabled application interaction sequence

In this section, the communication steps between Dynamix and the STAGE Manager are discussed. The interaction sequence between the two are shown in Figure 4.8. The Interaction Manager triggers and establishes the connection with the Dynamix framework. Upon connection, it receives all Dynamix related events as long as the Dynamix connection remains alive. For instance, the received events involve Dynamix activation events, deactivation events, and plugin installation events. Next, the Interaction Manager requests the ability profile from the Ability Manager. Moreover, it serializes the application's interaction profile, using its profile builder module.

The Interaction Manager requests Dynamix for all accessible plugins. Accessible plugins are those available in the local or network repositories. The Dynamix management interface allows the user to configure and link to trusted plugin repositories. Those repositories are then used for the plugin discovery process by Dynamix. Once the list is received, the Interaction Manager identifies all accessible IPs and ignores all other context plugins. Plugins that implement the required the "IPluginInfoObject" datatype

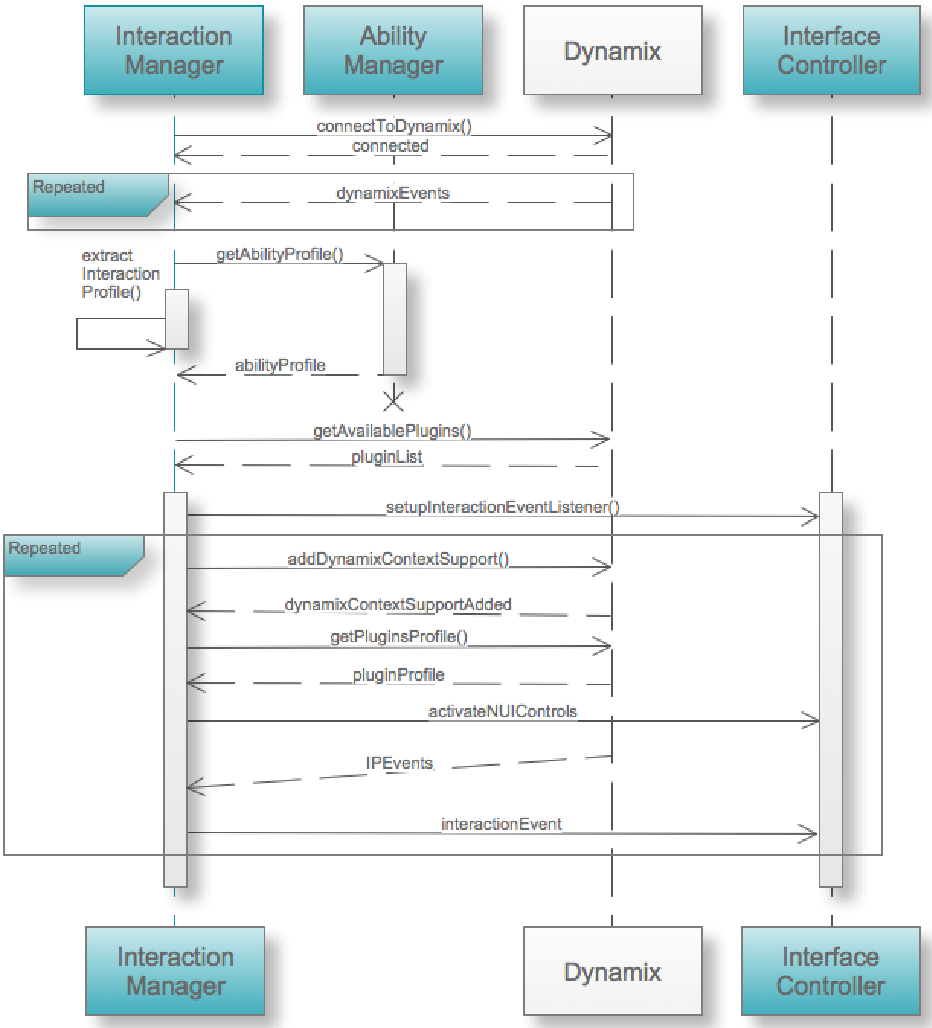


Figure 4.8.: Sequence Diagram for STAGE-enabled application

object proposed by STAGE are identified as IPs. Otherwise, the plugins are considered conventional Dynamix context plugins. Moreover, the Interaction Manager filters out all plugins that don't satisfy the interaction primitives required by the application. The plugin's supported interaction primitives appear as supported context datatypes by the plugin, hence are easily and directly distinguishable by Dynamix. Next, the manager subscribes to the "InfoObject" datatype for all IP satisfying the previous conditions. The "InfoObject" contains the essential Uniform Resource Identifiers (URIs) for the IP and its three profiles (movement, ability, and interaction), which are used later for filtering adequate IP in a given context. It then waits for receiving an event with the request plugin details. This even is fired only when the plugin is activated (i.e., upon the availability of the required hardware). While getting accepted subscriptions, the manager requests the plugin's ability and interaction profiles, in order to build a map of IPs and physical abilities required for each (Listing 2).

In order to set the ground for the Ensemble Engine, the Interaction Manager sends Dynamix subscription requests to the available IPs that satisfy the physical abilities of the user (Listing 3). This information forms the core base for the Ensemble Manager's matching algorithm. Accordingly, the Interaction Manager requests the Interface Controller to activate those GUI elements that are possible to be controlled by the available interaction resources (i.e., IPs). Other GUI elements, not supported by the available interaction resources, are not activated for NUI interactions, but can be still used conventionally (using the conventional touch interface).

```
if(primitivesFound){
// Register for plugin info
config = new Bundle();   // rest from previous operations
config.putString(ContextSupportConfig.REQUESTED_PLUGIN,
                 pluginInfoItem.getPluginId());
config.putString(ContextSupportConfig.CONTEXT_TYPE,
                "de.itm.STAGE.nui.plugin.info");
res = dynamix.addConfiguredContextSupport(dynamixCallback, config);
   if (!res.wasSuccessful())
      Log.w(TAG, "Call was unsuccessful! Message: "
       + res.getMessage() + " | Error code: "
      + res.getErrorCode());
}
```

Listing 2: Requesting IP information (i.e., Interaction Profile and Ability Profile)

At the end of interaction sequence, the STAGE Manager is able to receive modeled interaction events from the various IPs. The triggered interaction primitives are then sent to the Interface Controller to perform the required interaction tasks. It is important to know that multiple plugins may deliver the same interaction primitive. In this case, the Ensemble Engine may decide on which plugin should be used according to the best match with the user's physical abilities.

4.6. Performance evaluation

This section presents the results of the performance evaluation of the STAGE IP implementation. In their work, Carlson and Schrader [33] evaluated the Dynamix platform in terms of the processing power required and the heap characteristics during the most common operations, such as loading context

```
for (Primitive p : pluginInteractionProfile.getPrimitivesList()){
   int pluginHasContextType = isPrimitiveMatchAvailable(contextTypes, p,
   pluginInfoItem.getPluginId());
   Log.i(TAG, "**** Plugin Status ****: "
   + pluginInfoItem.getInstallStatus().toString() + " |"
   + pluginInfoItem.isEnabled());
   if((pluginInfoItem.getInstallStatus().compareTo
   (PluginInstallStatus.INSTALLED) == 0) && (pluginHasContextType != -1) ){
      primitivesFound = true;
      Log.i("STAGEDiscovery", "adding context "
      + contextTypes.get(pluginHasContextType) + " support for : "
      + pluginInfoItem.getPluginId() );
      Log.i("STAGEDiscovery", "adding context "
      + "de.itm.STAGE.nui.plugin.info" + " support for : "
       + pluginInfoItem.getPluginId() );
      config = new Bundle();
      config.putString(ContextSupportConfig.REQUESTED_PLUGIN,
      pluginInfoItem.getPluginId());
      config.putString(ContextSupportConfig.CONTEXT_TYPE,
      contextTypes.get(pluginHasContextType));
      res = dynamix.addConfiguredContextSupport(dynamixCallback, config);
      if (!res.wasSuccessful())
         Log.w(TAG,"Call was unsuccessful! Message: "
         + res.getMessage() + " | Error code: "
         + res.getErrorCode());
      }
}
```

Listing 3: STAGE Interaction Manager subscribing to IPs that satisfy the application's required interaction primitives

plugins and context scanning. Nonetheless, their evaluation results did not include information on plugin's performance in terms of responsiveness and realtime context delivery.

Information about plugins behaviour in high demand context modeling and delivery is very important in our problem domain, particularly because IPs facilitate and model user interactions. Therefore, a number of evaluation tests were performed to identify the performance of IPs in terms of the number of simultaneous plugin modeling, eventing frequency, eventing throughput, and eventing delay.

Herein, the used experimental setup and the evaluation results are presented.

4.6.1. Experimental setup and procedure

For the evaluation, a Galaxy Nexus (GT-I9250) smartphone with a 1.2 GHz dual-core ARM Cortex-A9 processor and 1 GB of internal memory was used. The device runs Android 4.2.2 (Jelly Bean) and Dynamix 0.0.59. To avoid external influences, the standard manufactory settings with no external applications installed were used. Moreover, both localization and networking services and capabilities (WiFi and Bluetooth) were deactivated during the evaluation. This device was selected in particular as an average off-the-shelf smartphone with the standard unmodified Android operation system.

For logging the performance, an external monitoring laptop (Macbook Pro with a 2 GHz Intel Core i7 processor and 8 GB memory) was used. The Android Debug Bridge (adb) was used to capture and filter out the Android console logging messages and store all relevant logs for an offline evaluation analysis.

The Android Debug Bridge (adb) is part of the Android Developer Tools (ADT).

This evaluation was intended to evaluate the eventing performance of IPs under different conditions. The main performance metrics focused on were the eventing throughput and delay, while changing the number of IPs and eventing frequency.

To reach the evaluation targets, an apparatus was implemented, which involved five identical configurable IPs (a configurable eventing rate and eventing trigger) and a simple application to consume the plugins' events. The performance IPs are autonomous plugins that perform context interactions autonomously in the background and broadcast interaction events accordingly. A performance IP only sends simple interaction events of type "PerformancePrimitiveInfo", which includes a string presenting the context type ("de.itm.stage.nui.primitives.performance"), a date object presenting the events's timestamp, a string presenting the event value (timestamp with event ID), and a string presenting the technology used for the plugin.

The performance application contained a simple interface to allow the experimenter to control the evaluation runs. The experimenter was able to set the required plugins, eventing frequency, number of eventing rounds, and the start of the test runs.

The experiment procedure included five steps. In each step, the number of plugins involved in the test run is increased by one (Step 1 included 1 IP, Step 2 included 2 IPs, etc). Each of the aforementioned steps contained five different rounds for each of the involved IPs. Each round was intended to test on of five eventing frequency levels (1,000 Hz, 100 Hz, 20 Hz, 10 Hz, and 5 Hz) and consisted of sending 1,000 events. After each round, a log file was labeled and saved. Additionally, the adb log history was cleared. After each of the steps, the device was restarted and the plugins were freshly installed.

The total number of events in each of the experimental steps is shown in the following list:

- Step1: 1 IP x 5 Frequency Levels x 1,000 Event = 5,000 Event.
- Step2: 2 IPs x 5 Frequency Levels x 1,000 Event = 10,000 Event.

- Step3: 3 IPs x 5 Frequency Levels x 1,000 Event = 15,000 Event.

- Step4: 4 IPs x 5 Frequency Levels x 1,000 Event = 20,000 Event.

- Step5: 5 IPs x 5 Frequency Levels x 1,000 Event = 25,000 Event.

4.6.2. Evaluation results

In this section, the results of the evaluation are presented. Table 4.1 shows an overview over the completed testing rounds. The table shows the average delay for each round (for each involved IP). Successful events are those received with delay less than the round's eventing period (i.e., before the next scheduled event).

		Delay (ms)				
		Step 1	**Step 2**	**Step 3**	**Step 4**	**Step 5**
IP#1	1000 Hz	177*	2979*	-	-	-
	100 Hz	5	32*	595*	3423*	-
	20 Hz	6	6	6	7	8
	10 Hz	9	7	7	7	8
	5 Hz	8	10	13	9	13
IP#2	1000 Hz		2724*	-	-	-
	100 Hz		31*	3595*	3595*	-
	20 Hz		6	7	7	8
	10 Hz		7	7	7	7
	5 Hz		11	10	10	10
IP#3	1000 Hz			-	-	-
	100 Hz			593*	3328*	-
	20 Hz			6	7	8
	10 Hz			7	8	7
	5 Hz			11	12	13
IP#4	1000 Hz				-	-
	100 Hz				3445*	-
	20 Hz				7	8
	10 Hz				7	7
	5 Hz				10	10
IP#5	1000 Hz					-
	100 Hz					-
	20 Hz					8
	10 Hz					7
	5 Hz					10

Table 4.1.: Eventing average delay for each tested IP in all experimental steps (each step consists of five rounds and each round consists of 1,000 events)

The table shows that the maximum eventing frequencies (1,000 Hz and 100 Hz) were not achieved (except for 100 Hz in step1). Unachieved frequencies were caused by either uncompleted experimental rounds (marked in the table as "-") due to Dynamix crashing or error exceptions; or very low received successful events with less than 99% of the throughput (delay values marked with "*") as in 100 Hz in step2 for #IP1 and #IP3. Hight eventing frequencies increase the size of the Dynamix event stack, this behaviour results into crashing Dynamix once the heap max was reached. This results correspond to the Dynamix performance evaluation in [33].

The results indicates that a high throughput is only achieved at 20 Hz, 10 Hz, and 5 Hz regardless the number of plugins. Generally, increasing the number of parallel active plugins increases the delay (although not too significant in many rounds). The table also shows an acceptable interaction response time (6 ms - 13 ms) for the three aforementioned frequencies. This delay is well below the limit of 100 ms, where the interaction delay is considered instant for the system's users [123]. A real IP will certainly require additional delay for processing and modeling context data into interaction events. Nonetheless, the current delay performance shows that additional delay in real scenarios will most likely to stay within an acceptable margin.

The delay spikes appeared in the table for the 100 Hz eventing frequency (in step 3 and step 4 for all plugins) are mainly caused due to memory overload and memory management operations (especially garbage collection processes).

The rest of this section will present the identified behaviour of plugins in each step. Both 1,000 Hz and 100 Hz are not depicted in the reported figures (except for Figure 4.9 for 100 Hz) as both frequencies were not achieved as reported above. Herein, we provide a closer look at the eventing delay evaluation for each step:

- Step1: Figure 4.9 illustrates the eventing delay for one plugin based on various frequencies (100 Hz, 20 Hz, 10 Hz, and 5 Hz). Small variations in delay were identified due to changing frequency, but with no clear correlation. The average delay amongst all rounds was 7 ms. The spikes appearing in the figure are identified due to Dynamix garbage collection, which affected all frequency levels. Nonetheless, the severity of those spikes is not critical. All spikes are registered below 25 ms, which is well below the acceptable delay for all frequencies.

- Step2: Figure 4.10 shows an acceptable average delays of 6 ms, 7 ms, and 11 ms for the three frequency levels (20 Hz, 10 Hz, and 5 Hz) respectively. The results indicate that the delay decreases by increasing frequency but at the same time the tolerance for delay is inherently higher for low frequencies. This mainly due allocating more computing resource for more demanding operations in Android. Nonetheless, this is not tolerated for very high frequencies (e.g., 100 Hz and 1,000 Hz) as reported previously.

- Step3: Likewise, Figure 4.11 shows an acceptable average delay of 6 ms, 7 ms, and 12 ms for the three frequency levels (20 Hz, 10 Hz, and 5 Hz) respectively.

- Step4: Similar behaviour was also recorded for this experimental step. Figure 4.12 shows an average delay of 7 ms, 7.5 ms, and 10 ms for the three frequency levels respectively.

- Step5: Figure 4.13 shows also an acceptable performance for all plugins with an average delay of 8 ms, 7 ms, and 11 ms for the three frequency levels respectively.

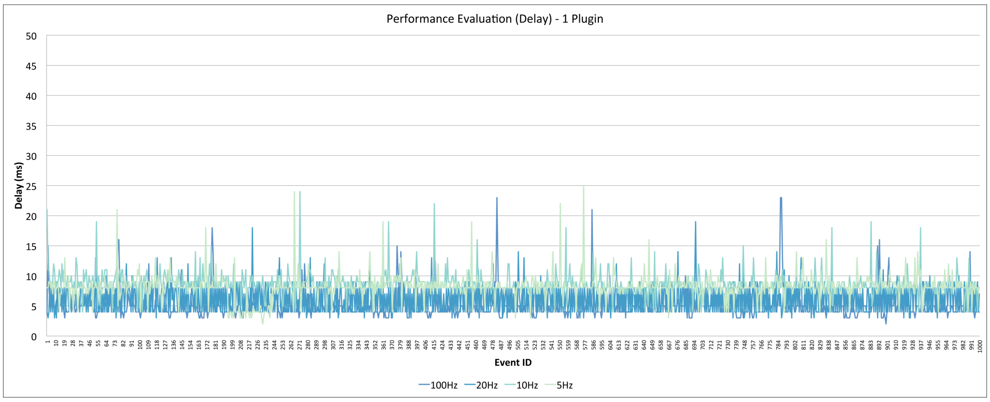


Figure 4.9.: Eventing delay evaluation (Step 1)

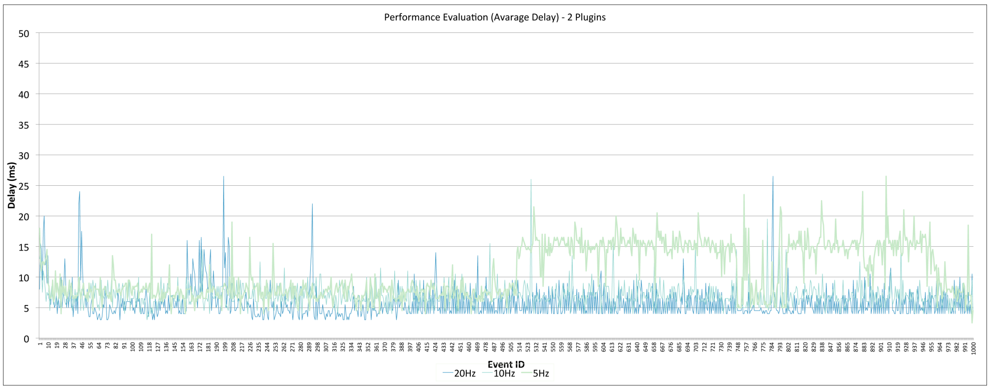


Figure 4.10.: Eventing delay evaluation (Step 2)

The evaluation results presented in this section show that increasing number of plugins is certainly a scalability issue for large scale deployment. However,

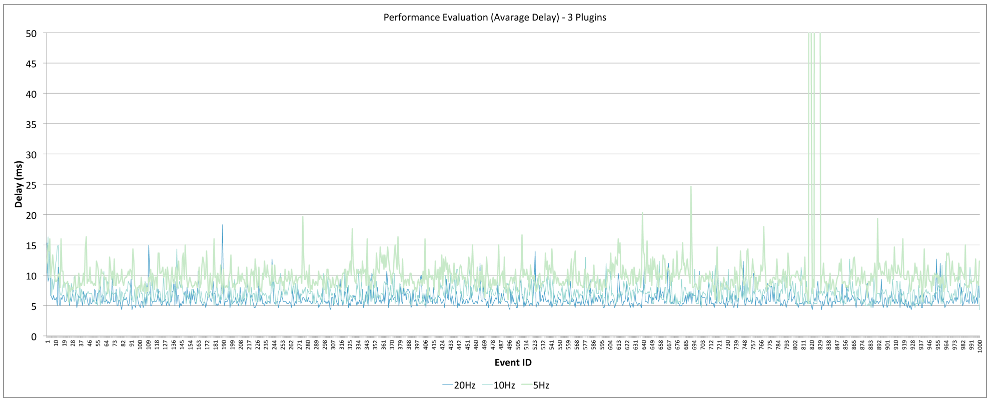


Figure 4.11.: Eventing delay evaluation (Step 3)

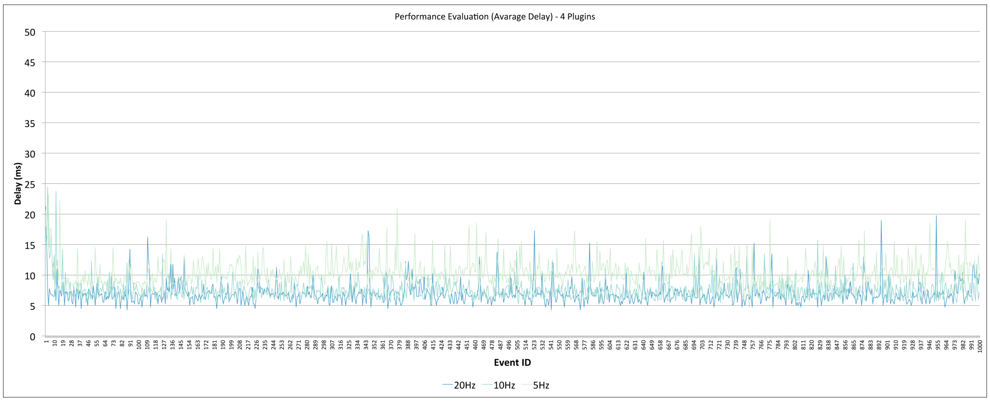


Figure 4.12.: Eventing delay evaluation (Step 4)

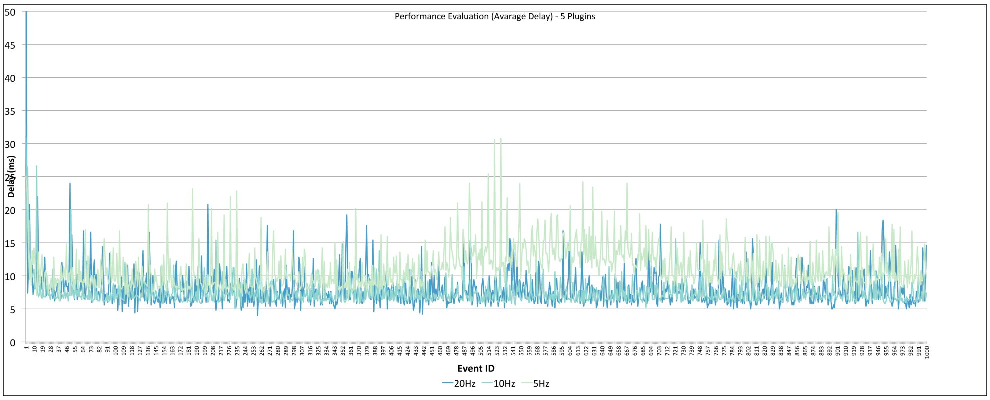


Figure 4.13.: Eventing delay evaluation (Step 5)

the responsiveness of the system to user interactions, in our current implementation, should not be greatly affected as IPs should send readily modeled interaction primitive events to applications, where up to 20 Hz interaction events per plugin seems realistic with an acceptable response time (6 ms - 12 ms).

4.7. Interaction Plugin examples and showcases

Herein, a few IP examples implemented based on our approach are presented. The interaction techniques for those plugins depend either on our own previous work, on techniques presented by other researchers, or by using available gesture libraries or frameworks. Developing the core acquisition and modeling algorithms for interaction techniques is out of the scope of this work.

4.7.1. 3Gear hand motion IPs

We appreciate and acknowledge 3Gear Systems for issuing us an academic license free of charge for the period of this work.

A number of hand motion based IPs, based on the 3Gear DevKit[3], are built. The 3Gear system provides various detection techniques for hand movements to control GUI on PCs or laptop machines. By developing STAGE IP, STAGE-enabled applications, running on mobile devices, are able to use the same gestures for interaction.

The 3Gear system is based on using different types of depth camera sensors for the detection. In our setup, we have deployed the Asus Xtion PRO Live developer solution[4]. We also use the OpenNI SDK[5] (OpenNI 2.2.0.33 Beta(OS X)) that facilitate the base drivers, libraries, and interfaces required.

In this section, we present two IPs implemented based on the 3Gear kit and their implementation workflow:

1. **Design and implement IP**: First, the interaction developer decides to build two IPs based on the left and right hand pinch gestures. Next, the supported interaction primitives for each IP should be defined. In this example each IP delivers a single interaction primitive based on different hand movements. The 3Gear right hand pinch plugin delivers a selection interaction primitive. The 3Gear left hand pinch plugin delivers a position interaction primitive. Next, the IP were named (3Gear Right Finger Pinch Selection and 3Gear Left Finger Pinch Position) after the kit's name, the gesture they present, and the interaction primitive they support. Finally, the core logic of the plugin including the connection to the 3Gear system is implemented and tested.

[3] http://threegear.com/index.html, accessed on 15.03.2014
[4] http://www.asus.com/Multimedia/Xtion_PRO/, accessed on 15.03.2014
[5] http://www.openni.org/openni-sdk/, accessed on 15.03.2014

2. **Create IP bundle**: In this step, the core logic of the plugin is implemented as part of a Dynamix bundle. This requires the developer to implement all required interfaces and methods from Dynamix. Moreover, the datatypes supported by the plugin are defined as well. The 3Gear right hand pinch plugin delivers a selection interaction primitive (de.itm.stage.nui.primitives.selection) as well as the default information datatype (de.itm.stage.nui.plugin.info). The same process applies to the left hand pinch plugin. The plugins are then exported as a standalone deployable bundle. This process also includes defining the plugin's description document (shown in Listing 5) which is used by Dynamix plugin discovery process.

3. **Define STAGE profiles**: This step requires the developer to define the movement, ability, and interaction profiles for the two plugins. The 3Gear right pinch interaction requires the user to extend the right arm at a 90-degree angle to the rest of the body pointing forward with the the palm of the hand pointing to the left and the hand naturally curved. Pinching is registered when the user's fingers tips touch each other for short time. Similarly, 3Gear left hand pinch plugin requires the user to extend the left arm at a 90-degree angle to the rest of the body pointing forward with the palm of the hand pointing to the right and the hand naturally curved. The full illustration of the movement profiles for the left and right hand pinch plugins are presented in section 6.4.3. The Interaction Profile is also defined for both plugins to reflect the interaction primitives supported. For instance, Listing 4 shows that a single interaction primitive (de.itm.stage.nui.primitives.selection) is supported by the 3Gear right hand pinch IP. Likewise, the ability profiles for the IPs are defined as in section 6.5.1.

4. **Publish IPs to the network repository**: This step, the IP bundles and the description documents are uploaded to the STAGE network repository. Accordingly, the plugins become accessible to the user. Figures 4.14 shows the aforementioned plugins as part of the Dynamix Plugin list (before and after the installation).

The utility of those plugins can be widely described according to the application. Each plugin may be associated with an interface component to control. In section 4.7.2, we describe the use of the 3Gear right and left hand pinch plugins as part of an interactive scenario to control the lighting of a smart room.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<appInteractionProfile>
 <primitives>
  <primitive>
   <type>de.itm.STAGE.nui.primitives.selection</type>
   <dynamixPluginDetails>
    <pluginId>de.itm.STAGE.interactionplugins.threegear.selection.rightfingerpinch
    </pluginId>
    <dynamixPluginType>AUTONOMOUS</dynamixPluginType>
   </dynamixPluginDetails>
  </primitive>
 </primitives>
</appInteractionProfile>
```

Listing 4: Interaction Profile for 3Gear right hand pinch IP

4.7.2. The AmbientRoom application

The Art-net Light Controller plugin is also used as part of our work on the AmbientWeb project [30]

The AmbientRoom application was implemented as a simple demonstration of use of IP by interactive applications in practice. The application's main goal is to control the ambient lighting of a smart room. The application relies on our implementation of the Art-Net Light Controller plugin to dim and change the light of the room according to the user preferences. The plugin uses the Art-Net protocol and enables sending DMX512 data for controlling lighting equipment over the Internet Protocol networks. The Art-Net plugin is a conventional Dynamix context plugin that acts as an actuator in this scenario, extensive discussion about this type of plugins is covered in [30]. Therefore, it should not be mixed with the IP concept discussed in this chapter.

As part of our setup for this application, we have deployed the "ArtNet-LED-Dimmer 4" light dimmer from DMX4ALL[6], to control a five meter LED RGB stripe. The hardware used for this scenario is shown in Figure 4.15.

First, we introduce the main GUI elements of the AmbientRoom application in Figure 4.16, before we analyze the interactivity of the application in Figure 4.17 and Figure 4.18. The application offers the user the ability to control the connection (i.e., connect and disconnect) with Dynamix (2), access the STAGE Panel for a detailed view on the used IPs and interaction profiles (3), and access a help view (4). The functions are accessible using three menu items at the top left corner of the screen respectively (from left to right). Moreover, the GUI contains four buttons to increase the room's light intensity level (5), reduce the light intensity level (6), change the light color (7), and

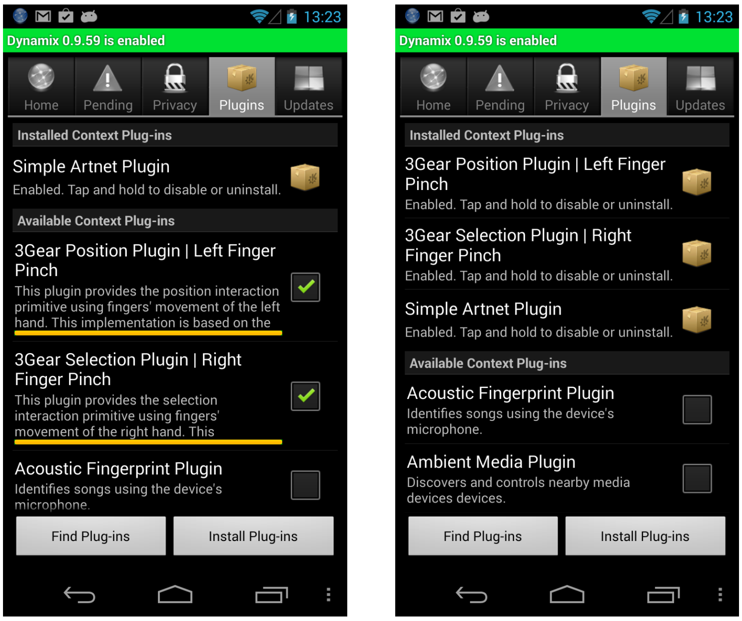
[6] http://www.dmx4all.de, accessed on 15.03.2014

```xml
<?xml version="1.0" encoding="UTF-8"?>
<contextPlugins
 version="1.0.0"
 xmlns:android="http://schemas.android.com/apk/res/android">
<contextPlugin
 metadataVersion="1.0.0"
 repoType="simple-file"
 id="de.itm.STAGE.interactionplugins.threegear.selection.rightfingerpinch"
 pluginVersion="1.1.0"
 pluginType="AUTONOMOUS"
 provider="STAGE"
 platform="android"
 minPlatformVersion="2.1"
 minFrameworkVersion="0.9.47"
 requiresConfiguration="false"
 hasConfigurationView="false"
 runtimeFactoryClass=
 "de.itm.STAGE.interactionplugins.threegear.selection.PluginFactory">
 <name>3Gear Selection Plugin | Right Finger Pinch</name>
 <description>This plugin provides the selection interaction primitive using
 fingers' movement of the right hand. This implementation is based on the
  3gear sdk.</description>
 <supportedPrivacyRiskLevels>
  <privacyRiskLevel name="LOW">This plugin does not use or collect any personal data.
   </privacyRiskLevel>
  </supportedPrivacyRiskLevels>
  <supportedContextTypes>
   <contextType>de.itm.STAGE.nui.primitives.selection</contextType>
   <contextType>de.itm.STAGE.nui.plugin.info</contextType>
  </supportedContextTypes>
  <featureDependencies/>
  <permissions/>
  <installUrl>dynamix/de.itm.STAGE.interactionplugins.threegear.selection
  .rightfingerpinch_1.1.0.jar</installUrl>
  <updateUrl></updateUrl>
  <updateMessage priority="OPTIONAL"></updateMessage>
 </contextPlugin>
</contextPlugins>
```

Listing 5: Context Plugin Description (CPD) for the 3Gear right hand pinch IP

turn the room's light on and off (8) respectively. Moreover, the application shows the name and an icon of the connected light dimmer controller (9).

The Figure 4.17 (1) illustrates the AmbientRoom application launch icon on the user's home screen. The Figure 4.17 (2) shows the application's main GUI elements in their disabled state and they will remain in this state until an adequate light controller plugin is available. In the background, application identifies three required plugins for its utility. The application requests a runtime deployment of the Art-Net Dynamix context plugin. Upon a successful deployment of the Art-Net context plugin (seen in Figure 4.17 (3)), GUI elements (i.e., buttons) to control the room's light are activated.



(a) IPs while installation (b) IPs after installation

Figure 4.14.: 3Gear IPs appear in Dynamix plugin control interface

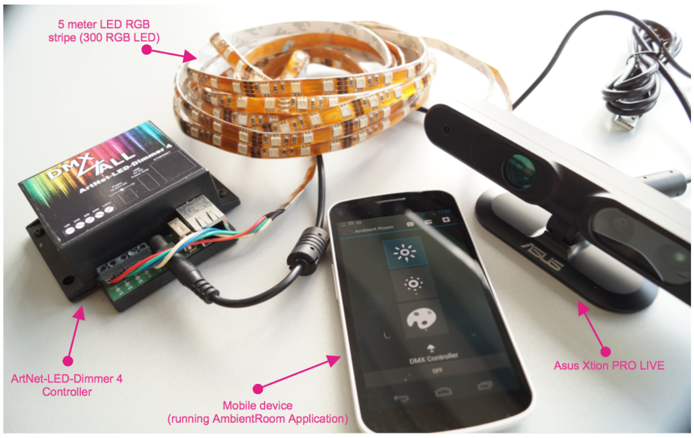


Figure 4.15.: The minimal set of hardware used for AmbientRoom demonstration scenario

Conventionally, the user can interact with the GUI using the touch enabled interface on the screen. But as the AmbientRoom application is a STAGE-enabled application, the application is able to utilize NI events for controlling the interface, based on the availability of IPs detected by STAGE. The AmbientRoom application requires particularly two interaction primitives, namely selection and position, as shown in the application's interaction profile (Listing 6).

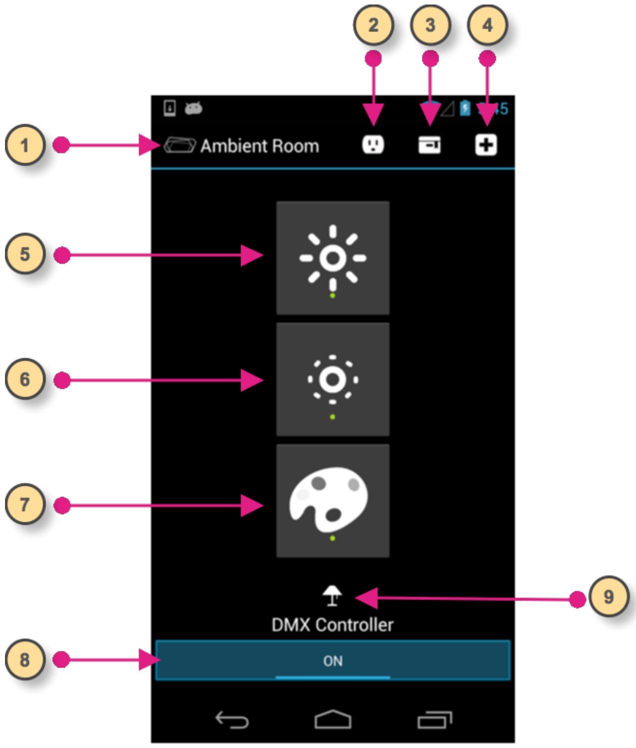


Figure 4.16.: AmbientRoom application main GUI

STAGE recognizes the required interaction primitives and accordingly will send Dynamix context type requests for the required types, but only when the Art-Net plugin is already activated as illustrated in Figure 4.17 (3). Once the context types are received as in Figure 4.17 (4), the user will be able to control the application using NUI interaction styles. In the AmbientRoom scenario we advocate the use of the 3Gear hand motion IPs for selection and positioning as an example, but other plugins that support the requested interaction primitives may be equally used as well. Accordingly, relevant interface components are marked with an indicator (i.e., visible tinted dot below the button's icon) as shown in Figure 4.17 (4).

Figure 4.18 demonstrates the NUI style interactions with pinch gestures enabled by the deployed IPs. Figure 4.18 (1) shows the NUI-enabled buttons. Using the pinching gesture with the left hand, the user positions the focus on the control of choice (each pinch will position the focus on the next button) as illustrated in Figure 4.18 (2). Using the right hand pinch, the user is able to select (i.e., push) the button in focus as illustrated in Figure 4.18 (3).

Figure 4.19 illustrates the different functionality offered by the STAGE Panel that is responsible to preset the user with a detailed view on the interaction profiles and the used IPs. The IP list (shown in Figure 4.19(a)) presents the

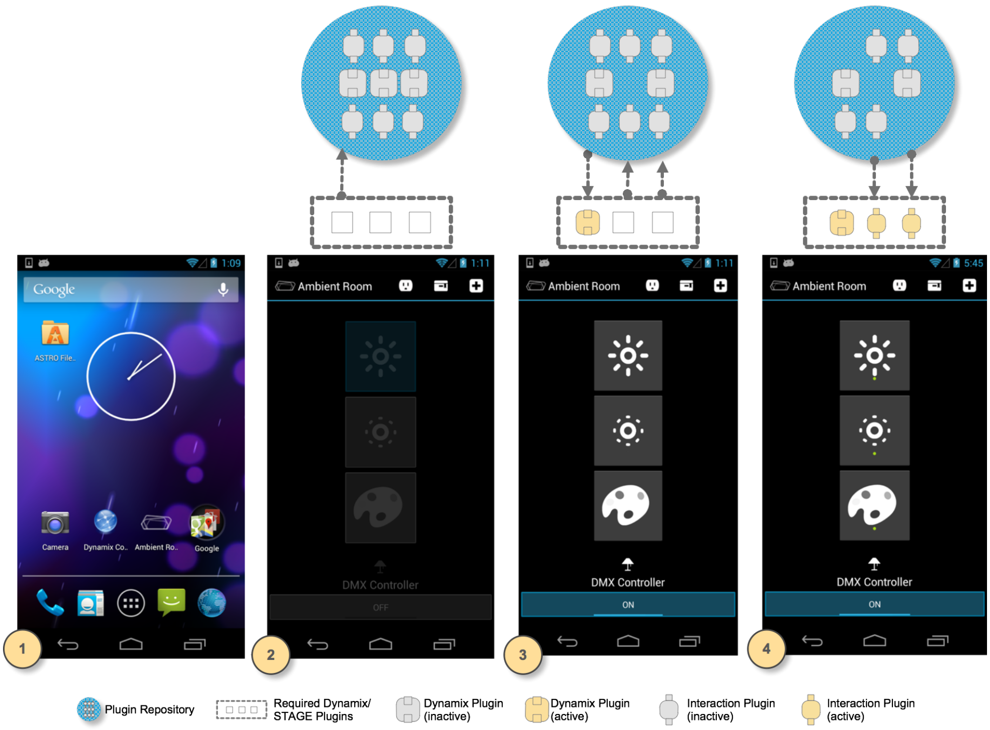


Figure 4.17.: AmbientRoom application (screenshots)

```
<?xml version="1.0" encoding="UTF-8"?>
<appInteractionProfile>
    <primitives>
        <primitive>
            <type>de.itm.STAGE.nui.primitives.selection</type>
            <dynamixPluginDetails>
                <pluginId></pluginId>
                <dynamixPluginType>AUTONOMOUS</dynamixPluginType>
            </dynamixPluginDetails>
        </primitive>
        <primitive>
            <type>de.itm.STAGE.nui.primitives.position</type>
            <dynamixPluginDetails>
                <pluginId></pluginId>
                <dynamixPluginType>AUTONOMOUS</dynamixPluginType>
            </dynamixPluginDetails>
        </primitive>
    </primitives>
</appInteractionProfile>
```

Listing 6: Interaction profile (the AmbientRoom application)

user with all used IPs by the application, as well as with a quick overview on

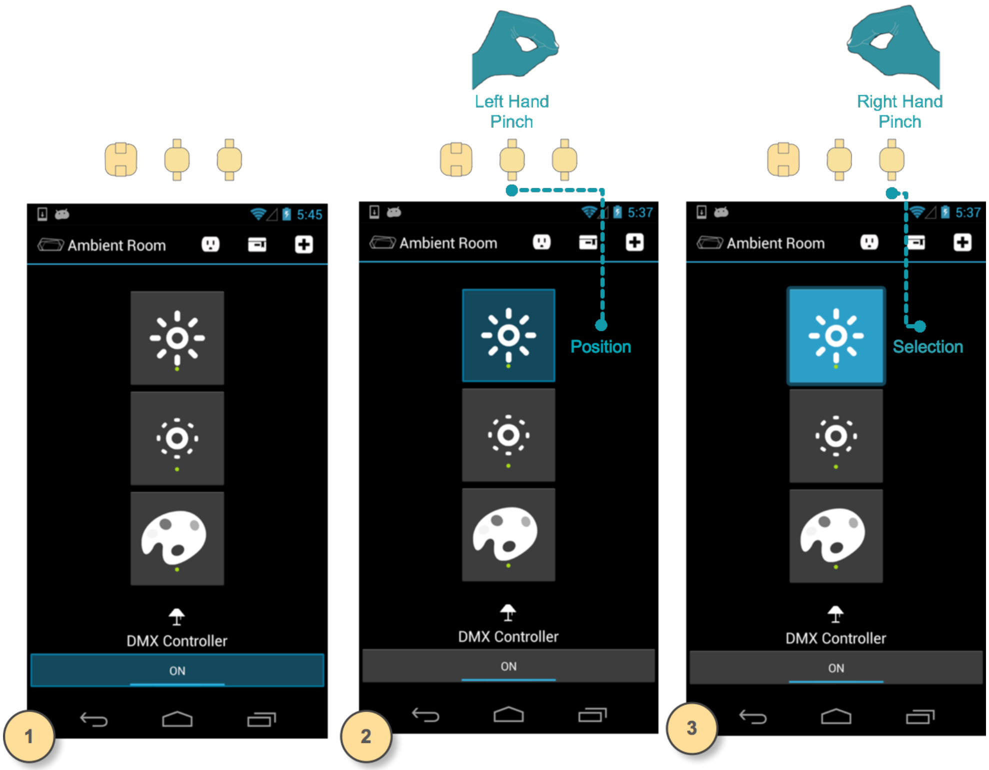
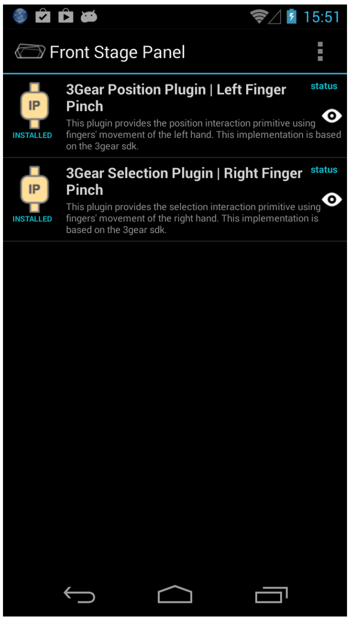
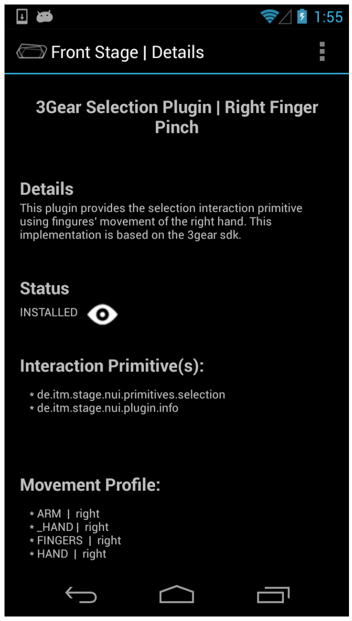


Figure 4.18.: AmbientRoom application NUI-enabled Interactions

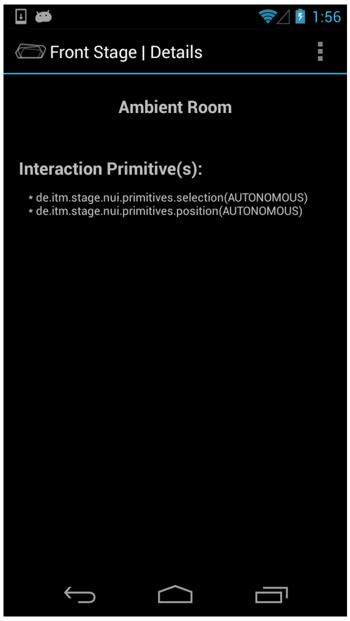
the plugins name, description, and status. The IPs offered in this list are offered by the Ensemble Engine based on the user's physical profile. Only IPs that satisfy the application's semantics (i.e., selection or position primitives) and match the physical abilities of the user are presented. Other plugins are filtered out. Currently, the Ensemble Engine does not offer the users a explicit manual control over the interaction possibilities, hence this list changes only to reflect changes in the user's ability profile. Allowing explicit editing of this list is envisioned in future releases. Figure 4.19(b) shows the IP detailed view, which presents the user with detailed information about the plugin. Most importantly, it shows information about the plugins's status, interaction primitives offered by the IP, and summery about the movement profile (involved body parts). Furthermore, the STAGE Panel offers the user with a view summarizing the application interaction profile (shown in Figure 4.19(c)).



(a) List of used IP and their status



(b) Detailed view on a selected IP



(c) The application's interaction profile view

Figure 4.19.: AmbientRoom application (STAGE Panel)

4.8. Conclusion

In this chapter, the deployment aspects of interaction techniques in ambient spaces are discussed, especially the richness and challenges of the dynamic component integration and adaptation in ambient computing research. Previously, the importance of deployment for the Interaction Ensemble concept was discussed. Despite its importance, this chapter revealed that deployment of NI and the composition of NI techniques into ensembles of interaction techniques are still relatively weak topics in HCI. To our best knowledge, the development and advancement in this direction are still very limited to few research efforts, as shown in this section.

Throughout the course of this chapter, the STAGE technical implementation of the IP concept was presented to deploy interaction techniques in ambient spaces dynamically at runtime. This type of deployment enhances the possibilities to adopting interactions in environments with changing context. The content presented in this chapter demonstrates the feasibility of the approach through an actual implementation and a small scale evaluation based on mobile devices.

Although the adoption of interactive applications running on mobile devices may appear limiting in terms of utility and scope, we have envisioned and used the mobile device as a personalized interaction hub to facilitate interactions in ambient spaces. With the AmbientRoom application, we have demonstrated how a mobile device became a center control and interaction hub to control the room's lighting based on hand gestures. In addition to this application, a few IP examples are presented and their use in the AmbientRoom application is discussed.

To validate the performance of Dynamix and the IP implementation, the chapter presents a short performance evaluation study. The study reveals various scalability issues. The increasing number of simultaneous plugins involved reduces the maximum reachable eventing frequency levels. Nonetheless, the current implementation handles an acceptable number of plugins, when measured against realistic scenarios specially with eventing frequency up to 50 Hz for 5 individual simultaneous plugins. We believe that increasing hardware capabilities and improving Dynamix eventing for high frequency plugins definitely contribute to enhance the performance.

5 A Review on Movement Documentation and Learning Practices for Interaction in NUIs

5.1. Introduction

Documenting NUIs is a very important part in this dissertation. Hence, this topic is covered extensively in two chapters. This chapter covers a detailed overview on interaction documentation and learning practices. On the other hand, Chapter 6 covers a detailed overview on movements and abilities documentation for NUI. Additionally, it presents the main design and implementation choices considered in our work.

Dozens of novel interaction techniques are proposed every year to enrich interactive eco-systems. The increasing role of the body and its movements presents new documentation challenges for long-term record-keeping, dissemination, and sharing of interaction techniques. In this chapter, a novel investigation of the community's applied documentation practices and the users' learning habits for movements in Kinetic interactions (especially multitouch- and motion-based interactions) is performed. The investigation includes analyzing a survey targeted at NUI designers and users; a large sample of recently published multitouch and motion-based interaction papers; and three new motion-based applications market initiatives. A fourth side of this investigation (documentation-related tools and languages) is covered in section 6.2. Although limited in scale, this investigation opens the door for important open research issues regarding the needs for new tools, guidelines, strategies, and systems that may improve those practices.

Most fields that relay on movements, such as dance and choreography, strive for higher in quality and more complete movements documentation [158]. An equivalent interest in movements documentation for interactions should be also considered within the HCI community. Documenting interactions' movements is vital for ensuring the availability to designers, developers, and scholars for study, analysis, reproduction, evaluation, audience-building, and publicity purposes. Likewise, documentation is important for ensuring the availability to users for learning and performance purposes. Norman [126] argued despite the wide adoption of gestures, they are neither natural nor

An overview on NUI is covered in Chapter 2 and the disability challenges for interaction is covered in section 2.2.5 easy to learn and remember. A simple command, like "raise your arm", may have very different interpretations and different aspects should be considered for correctly executing such a simple command, for instance movement direction, involved body parts, and timing information. Moreover, the complexity of interactions in NUIs becomes more challenging with an expanding user population and its diversity, with respect to age and physical abilities.

Documentation is also crucial for engineering, disseminating, deploying, and adapting multitouch- and motion-based interaction techniques and technologies, especially because of the increasing user heterogeneity (e.g., aging and demographic change), user mobility to unknown environments, and spontaneous construction of interactive environments in-situ at runtime. New approaches for building interaction systems, such as Interaction Ensembles, require an adequate interaction description for discovering, filtering, and composing technical components for interactions based on the user's physical context and abilities.

Interactivity challenges in ambient spaces are covered in Chapter 3

We believe that an investigation of this kind is essential to understand some of the challenges for engineering interactive systems in ambient spaces, especially because interactions are becoming increasingly dynamic, adaptive and multi-modal. In literature, interactions are formally described and modeled based on data description, state representation, event representation, timing, concurrent behaviour, dynamic instantiation, etc [122][67]. An extensive review on those approaches is out of the scope of our work. Herein, the term documentation is used to capture the way physical movements, required by an interaction technique, are defined and described by the interaction designer (i.e., developer). Documentation relates to the vital movement aspects of the interaction such as involved body parts, type of movement, timing and sequencing, and movement dynamics. Principally, documentation refers to any material, such as written clues, visual clues, animated clues, formal description models and languages, etc., used to describe or disseminate the developed interaction.

Against this background, better understanding and analysis of the practiced documentation habits of movements in interaction techniques are required, especially for correct execution of interactions by end users, the preservation of technique by designers, the accumulation of knowledge for the community, and the engineering of interactive systems.

Despite the relevance of this problem to the HCI community, we believe that it is not well investigated. The literature lacks reported studies on NUI designers' documentation habits and dissemination strategies. This is manifested by a limited set of available tools and standards, which will be discussed in section 6.2 "Documentation-related Tools and Languages". The goals of this chapter are to reveal parts of the practices to document NUI for preservation and sharing purposes as well as to highlight parts of the NUI users' learning habits.

In this chapter, three areas of our four-fold investigation approach are presented, namely (1) analyzing a tailored survey targeted at designers and users (covered in section 5.2); (2) analyzing a large sample of recently published multitouch- and motion-based interaction papers (covered in section 5.3); and (3) reviewing a number of multitouch- and motion-based application market places (covered in section 5.4). We have supported the data with a number of general observations to enhance the readability of the results. The observations are numbered and marked with an abbreviation to the corresponding section as shown in Table 5.1. Section 5.5 presents a general discussion of the results. Finally, section 5.6 briefly presents the main conclusions of the investigation.

The fourth side of this investigation (documentation-related tools and languages) is covered in section 6.2

Abbreviation	**Investigation Section**
D	Designer survey section
U	User survey section
P	Papers analysis section
M	Market places analysis section
T	Documentation-related tools and languages section

Table 5.1.: Study observation abbreviations

To the best of our knowledge, this work is the first to closely investigate this issue and to trigger an elaborate discussion revealing its existing and forthcoming problems. Herein, we aim to target the following questions:

- What are the NUI designers' most commonly applied documentation choices, most importantly, documentation frequency and media type of choice?

- Are good NUI documentation practices observed and followed by designers?

- What are the NUI users' most commonly applied learning habits and preferences?

- Is there any mismatch between the NUI designers' documentation practices and users' preferred learning practices?

- How are NUI documentation currently considered in motion-based application market initiatives?

5.2. Survey on Interaction documentation

The first step in our investigation was to capture a snapshot on the current most employed practices for movement documentation by carrying out an online survey. The survey aimed to partially characterize a number of designers' documentation practices and users' learning habits, including:

1. The adoption level and frequency of documentation practices and standards in design and development of NUI;

2. The designers' satisfaction with their practiced NUI documentation habits;

3. The needs for new documentation tools and methods;

4. The current and preferred learning practices, methods, tools, and media types;

5. The impact of documentation on learning NUI;

6. The match level between the available NUI documentation and the users' learning preferences; and,

7. The perceived importance of documentation for sharing, acceptance, user experience, and correctness.

The survey was split into a user section containing 9 multiple choice and likert scale questions and a designer section containing 11 questions. For redirecting the respondent to either of the sections, the respondent was explicitly asked to choose to be either a normal user or an interaction designer (i.e., designing/developing interaction techniques). The survey was bound to a maximum completion time of 3 minutes to maximize the number of voluntary participations. The survey included an introductory section on multitouch- and motion-based interactions in the context of NUIs, as well as the purpose of the survey.

The survey was distributed online through specialized HCI-related mailing lists (including BCS-HCI run by the British Computer Society Human-Computer Interaction Group[1]), ubiquitous computing mailing lists (including Ukubinet-announce run by the Imperial College London[2] and announcements@ubicomp.org[3]), university mailing lists (such as Luebeck

[1] https://www.jiscmail.ac.uk/cgi-bin/webadmin?A0=bcs-hci, latest access on 25.03.2014.

[2] https://mailman.ic.ac.uk/mailman/listinfo/ukubinet-announce, latest access on 25.03.2014.

[3] http://mail.ubicomp.org/mailman/listinfo/announcements_ubicomp.org, latest access on 25.03.2014.

University[4]), and social networks (i.e., Facebook, Twitter, and ResearchGate). The survey was open for participation for about 3 weeks (between 27.03.2013 - 16.04.2013).

A total of 332 anonymous individual responses were recorded, split into 267 users (80% of the total respondents) and 65 designers (20% of the total respondents).

5.2.1. Designer section

The designer respondents are split to 11 expert designers, 14 professional designers, 28 competent designers, 10 advanced beginners, and finally 2 novice designers. This categorization is based on an explicit self-assessment question about expertise. Although not explicitly distinguished in the survey, we believe that most of the responses relate, but not exclusively, to research academics. This is mainly due to the distribution channels of our survey, discussed in the methodology section.

We have applied Kruskal Wallis test to identify any statistically significant differences among expertise groups. Apart from the observation (D2), no statistical differences amongst groups were found.

The respondents were asked to indicate whether they are satisfied with their current documentation habits. As a response to this question, 57% of the designers responded positively and indicated that they are satisfied with their current documentation habits.

> Observation - **D1**:
>
> Small majority of interaction designers are satisfied with their current documentation practices.

Figure 5.1 shows how often the designer respondents document the designed interaction techniques, independent of form or documentation type. The figure reveals that the majority of the respondents practice documentation either sometimes (42%) or frequently (38%). Merely small minority of designers (14%) practice documentation always. Statistically significant difference was identified among expertise ($H(4) = 13.466$, $p = 0.009$) with a mean rank of 43.93 for proficient, 33.75 for competent, 32.91 for expert, 18.70 for advanced beginner, and 18.0 for novice designers. Higher mean ranks indicate a more frequent documentation practice.

[4]https://www.itm.uni-luebeck.de, latest access on 15.03.2014.

Observation - **D2**:

Only a small minority of designers practice documentation always.

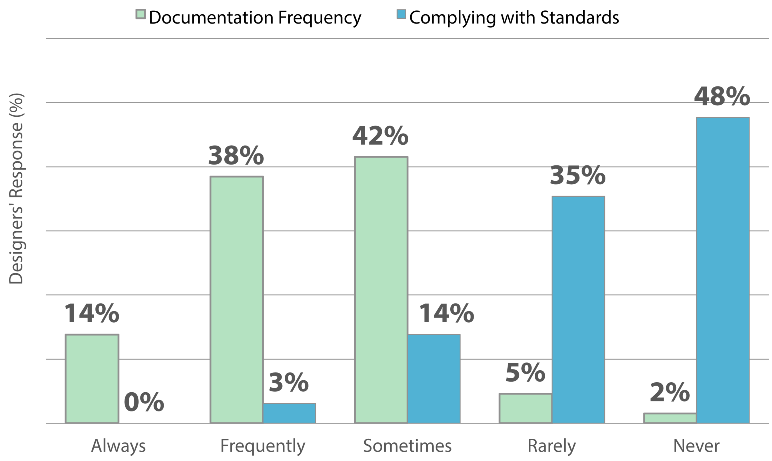


Figure 5.1.: Practicing documentation and complying with standards

One interesting aspect in this survey is to highlight the designers' habits to apply standard documentation approaches. To clarify these approaches, the survey contained a non-exclusive list as an example of available documentation approaches including Labanotation [77], GestureML[5], and Body Action and Posture Coding System [41]. Shown in Figure 5.1, the survey unveils that about half of the respondents never apply any documentation standards and merely a third did on rare occasions. Small number of respondents apply documentation standards either sometimes (14%) or frequently (3%).

Observation - **D3**:

The vast majority of NUI designers never or rarely apply documentation standards.

The respondents answered positively (66%) when asked whether there is a lack of documentation methods and tools available for them to use.

[5] http://www.gestureml.org/wiki/index.php/Main_Page, latest access on 20.04.2014.

> **Observation - D4:**
>
> The majority of NUI designers indicated a lack of NUI documentation tools and methods.

Another goal of the survey was to identify the dominant media types used by designers to document interaction techniques. The data marked as "Designed" in Figure 5.2 illustrates the distribution of used documentation media types by designers (data marked as "used" and "preferred" will be discussed in the "user section"). Text is the most used medium to describe and document interaction techniques. Still visual documentation records (i.e., pictures and sketches) follow next. Moreover, animated visual records come fourth. Additionally, audio and formal languages come even later with very low percentages.

> **Observation - D5:**
>
> Interactions for NUIs are mostly documented using text, pictures, sketches, and videos respectively.

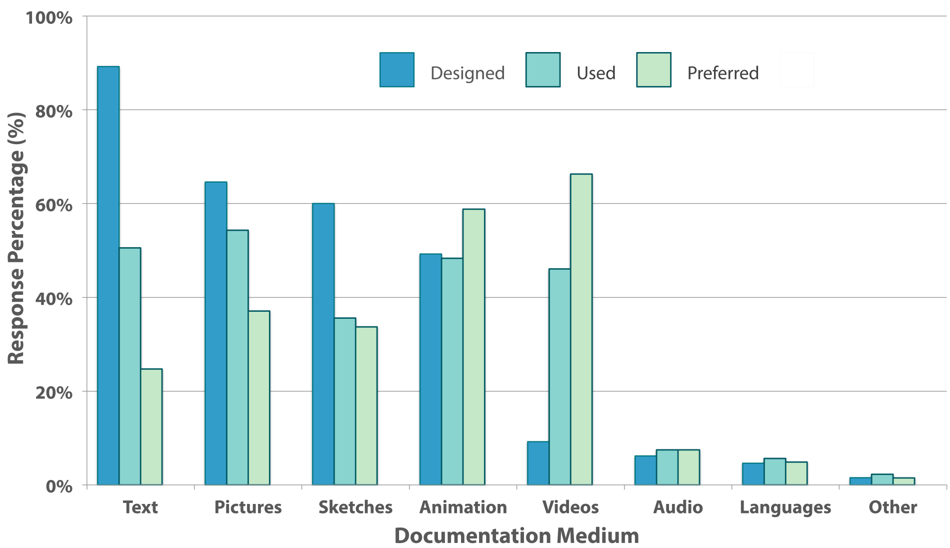


Figure 5.2.: Medium for documentation (used: by users, preferred: by users, designed: by designers)

Figure 5.2 also shows clearly that designers don't follow formalizations as a documentation media type.

Observation - D6:

Multitouch- and motion-based interactions are rarely documented using formalized languages.

Observation - D7:

The most ranked importance of documentation is acknowledged for sharing interactions, followed by user experience.

Figure 5.3 illustrates the designers' perceived importance of documentation for sharing, experience, acceptance, and correctness. The vast majority of responders scored documentation as a very important (45%) or an important (37%) factor for a successful sharing of interactions. Regarding user experience, the majority of respondents scored the documentation as an important (48%) or a very important (11%) factor respectively. Moreover, designers scored interaction documentation for user acceptance as very important (14%), important (40%), moderate (25%), and of little importance (18%). Merely 3% negatively scored documentation as unimportant for the user acceptance. Finally, the majority of respondents scored documentation as either an important (40%) or very important (26%) factor for the correctness of interaction execution. Approximately one third of the respondents scored documentation as moderate or of little importance for correctness.

5.2.2. User section

Multitouch- and motion-based interactions are currently accessible in a wide range of end user devices. The vast majority of user respondents (84%) use mobile phone, but other devices are used as well such as tablets (38%), game consoles (26%), interactive TVs (23%), touch surfaces (19%), and NUI controllers (18%).

Observation - U1:

All users demand multi-modality as a desirable requirement for interaction documentation.

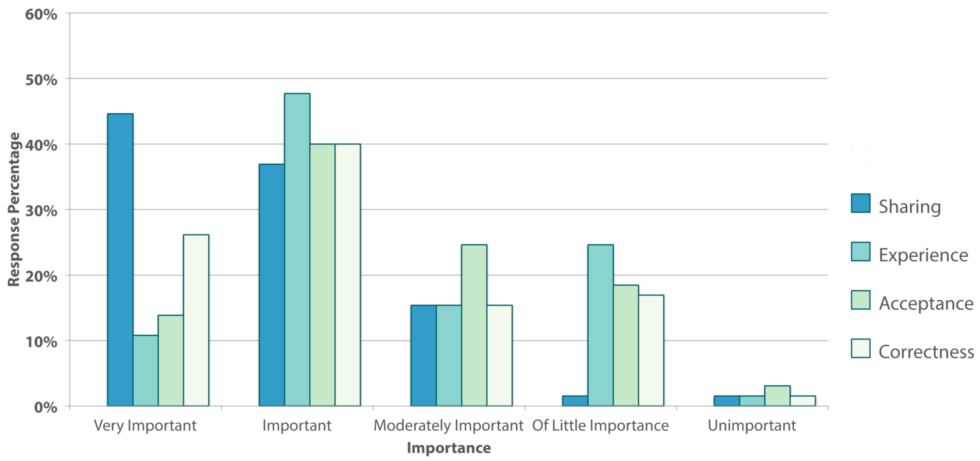


Figure 5.3.: The designers' perceived importance of documentation for acceptance, correctness, experience, and sharing of interactions

When asked whether interaction documentation should be made multimodal, users strongly agreed (49%) or at least agreed (41%) that multi-modality is a desirable requirement for interaction documentation. Multi-modality was described to the survey's participants as a a mixture of textual, audio, and visual media types used in combination to create interaction documentation (e.g., text and images, videos and text, etc.)

Observation - U2:

Finding well documented interactions positively correlates with the users' perception of knowing and using interaction features.

This observation is concluded by correlating "How often users find well-documented NUI" and "Users' confidence regarding knowing and using NUI available to them". Figure 5.4 shows that the more often users find well documented interactions, the more they are likely to know and use all available interaction features. All users who always find documentation at least agree to know and use all interaction features available on their devices. The vast majority of users (75%) who often find well documented interactions agree or strongly agree to know and use all available interaction features. This is followed by users who sometimes (54%) or rarely (42%) find well documented interactions.

Similar to the designer section, we were interested to find out documentation methods and media types used by users to learn new interactions.

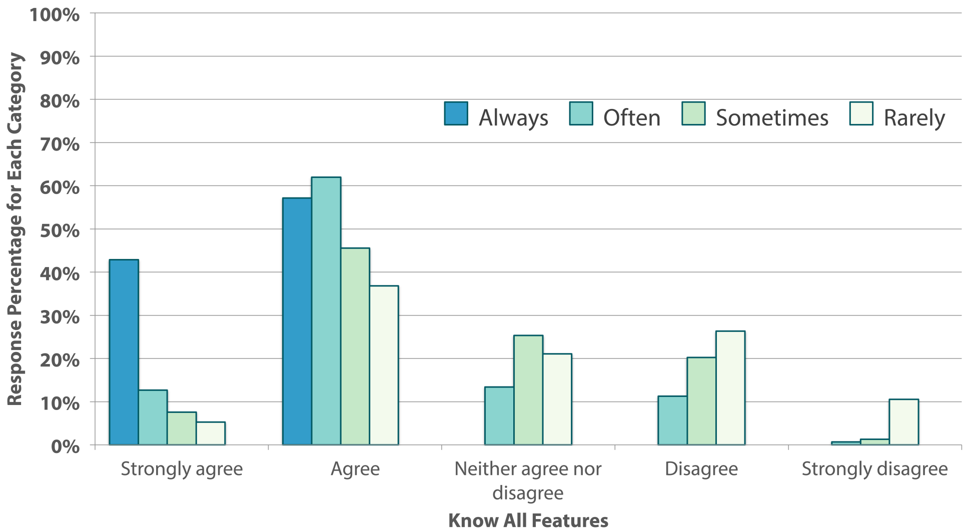


Figure 5.4.: Responding users - users' agreement on correctly using and knowing all interaction features categorized based on the frequency of finding well documented interactions.

Respectively, participants were asked to choose the most currently used and preferred methods and media types. The participants were able to choose multiple answers simultaneously, as learning may often involve the usage of multiple learning methods and media types. The data marked as "Used" and "Preferred" in Figure 5.2 illustrates the distribution of used and preferred documentation media types by users.

Observation - U3:

Users voted pictures, text, animation, and videos as the most currently used documentation media types.

Figure 5.2 reveals that the most used media types are pictures (54%), text (51%), animations (48%) and videos (46%). Sketches are reported by 34% of the respondents. Other media types such as audio, languages, etc. are reported by very small minority of users.

Observation - U4:

Users voted videos and animations as the most preferred documentation media for learning interactions.

Both videos (66%) and animations (59%) are reported to be the most preferred media types for documentation. Pictures are preferred by 37% of the respondents.

> **Observation - U5:**
>
> There is a mismatch between the most preferred and currently used media type.

Although text is currently used by about half of the responders, it is placed as one of the least preferred documentation media types (25%). Moreover, only 37% of the respondents preferred pictures, 17% less compared to the "used" percentage. In contrast, both videos (66%) and animations (59%) were scored as the most preferred media types respectively. The difference between "used" and "preferred" percentage is relatively small for sketches, audio, and languages.

We should point out that there is a noticeable mismatch between the reported "used" media type by users and the "designed" types by designers. This mismatch is clearly visible for the media types text, sketches, and videos. This mismatch may be due to our designer respondents, as we targeted mainly the research community in this part of our investigation. Clearly, text and sketches are two of main popular media types within this community (elaborated in the section 5.3). It is clearly visible that our design responders use far more text and sketches and far less videos. Other media types match the user respondents. Nonetheless, our data doesn't reveal or clarify the causes of this mismatch.

Figure 5.5 illustrates the distribution of various learning methods used and preferred by users. The methods include catalogs and manuals (provided usually by product manufacturers), individual learning by try and error, face to face learning from another user, learning from advertisements, and learning from interactive walkthrough wizards (increasingly used in commodity devices to aid the user to follow a step by step guiding or demonstration process).

> **Observation - U6:**
>
> Most users currently use "Try and Error" alone or combined with other methods to learn new interactions.

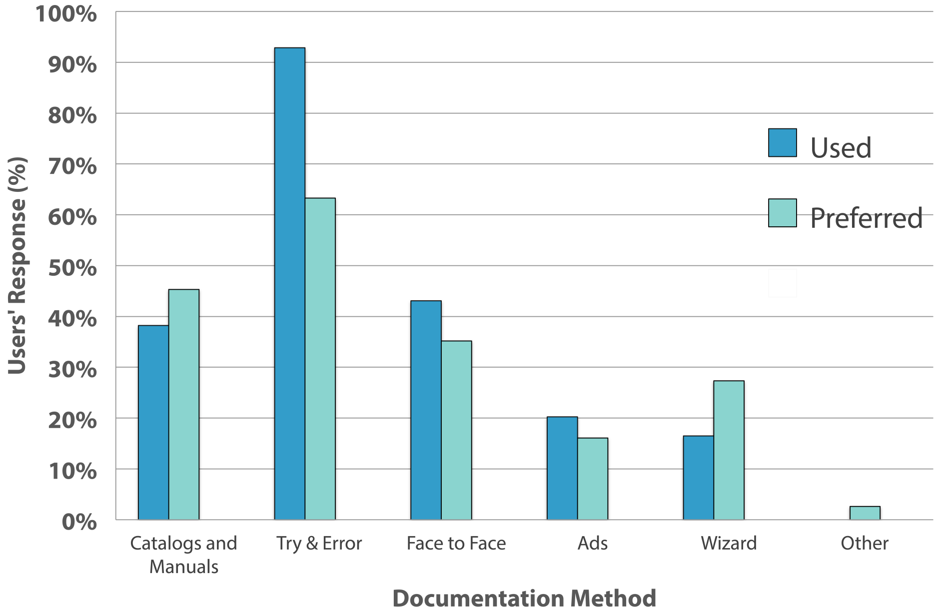


Figure 5.5.: Responding users - used vs. preferred documentation methods

Figure 5.5 shows that the majority of users (93%) rely on try and error to learn new interactions. Other methods are also used such as face to face learning (43%) and manuals (38%). Moreover, users reported the use of advertisement materials (20%) and step by step wizards (16%) as a learning source.

Observation - U7:

A noticeable mismatch between the most preferred and currently used methods for learning is visible.

Comparing the users' used and preferred learning methods in Figure 5.5, reveals a clear percentage mismatch for all methods, but exceptionally distinctive for "Try and Error". Although "Try and Error" remains as the most preferred learning method for the majority of users (63%), but with a staggering percentage difference (30%) to the used percentage (reported in U6).

5.3. Analyzing the interaction publications landscape

The second step in our investigation intended to capture a closer look at the published research work in the area of interaction techniques. We have decided to base our investigation on a collection of the most recent ACM published work under the ACM classification (H.5.2 Information Interfaces and Presentation: User Interfaces - Input devices and strategies) for the years 2012 and 2013 (until 22.08.2013), independent from the publication venue (ACM Conference on Human Factors in Computing Systems (CHI), ACM Symposium on User Interface Software and Technology (UIST), ACM SIGCHI Symposium on Engineering Interactive Computing Systems (EICS), etc.) Out of 518 total papers in this category, we manually coded and analyzed a total sum of 93 papers that matched one of two categories: (1) papers presenting novel interaction techniques; (2) papers applying or analyzing existing interaction techniques in various scenarios. Our filtering criteria excluded all none touch or none motion gesture papers (as considered out of the focus of this investigation), video papers (as those papers don't have enough space to cover the interaction technique and only convey very limited aspects of the work), and duplicated paper entries (if the same work was presented in multiple venues but with different contribution size, e.g., work-in-progress papers, short papers, full papers). In the case of duplication, the latest and longest contribution was considered. Our aim was not to conduct a complete and detailed review of all published papers. Instead, we aimed at providing a snapshot at the most recent published work as a living example of the current practiced documentation habits.

Our analysis and classification are based on the published paper and any corresponding material directly mentioned, linked, or attached with the published work (e.g., many published papers have also videos attached within the ACM library, or links to external resources). Other materials out of the aforementioned criteria were considered hidden and were not included in the study, such as in application help menus or offline accessible manuals.

The papers were coded based on four main aspects: *Type* - gesture types discussed in the paper including multitouch and motion gestures; *Still* - used still media types to document and describe the gesture including text, images, and sketches; *Animated* - used animated media types to document and describe the gesture including videos, animations, personal walkthrough, and onscreen walkthrough; and finally *Authoring* - reported or used authoring and documentation tools and formal languages. Our main goal of this analysis was to highlight general practices and habits rather than focusing on a particular paper title or the authors. Hence, we reference the reviewed papers by the unique identification key (ACM ID) instead of the papers' full title or author names.

Use		Authoring		Animated				Still			Type	
	ACM ID	Languages	Tools	Walkthrough (S)	Walkthrough (P)	Animation	Video	Sketches	Pictures	Text	Motion	Touch
40	2481423							⊙	⊙	⊙		⊙
41	2481424								⊙	⊙	⊙	
42	2468582							⊙	⊙	⊙	⊙	⊙
43	2468590							⊙		⊙		⊙
44	2468630				⊙			⊙		⊙		⊙
45	2468400									⊙		⊙
46	2468487							⊙		⊙		⊙
47	2491369				⊙				⊙	⊙	⊙	⊙
48	2465966								⊙	⊙	⊙	
49	2465961							⊙		⊙		⊙
50	2465972						⊙	⊙		⊙	⊙	
51	2491573				⊙	⊙		⊙		⊙	⊙	
52	2449452							⊙		⊙		⊙
53	2459248				⊙				⊙	⊙		⊙
54	2447609									⊙	⊙	
55	2460694									⊙	⊙	
56	2460702							⊙	⊙	⊙	⊙	
57	2388706							⊙	⊙	⊙	⊙	
58	2399091							⊙		⊙		⊙
59	2384945				⊙				⊙	⊙		⊙
60	2305305							⊙		⊙	⊙	⊙
61	2399103									⊙	⊙	
62	2168636				⊙		⊙			⊙	⊙	⊙
63	2393165				⊙					⊙	⊙	
64	2148191								⊙	⊙		
65	2379264								⊙	⊙	⊙	
66	2370265								⊙	⊙	⊙	⊙
67	2305297								⊙	⊙		⊙
68	2318032								⊙	⊙	⊙	⊙
69	2414572							⊙		⊙		⊙
70	2148190				⊙		⊙		⊙	⊙	⊙	
71	2254669				⊙				⊙	⊙	⊙	
72	2371599							⊙		⊙	⊙	⊙
73	2325644							⊙		⊙	⊙	
74	2414614								⊙	⊙	⊙	
75	2148142						⊙	⊙		⊙	⊙	⊙
76	2399104						⊙	⊙		⊙	⊙	⊙
77	2379644								⊙	⊙	⊙	⊙
78	2254582								⊙	⊙	⊙	
79	2350070								⊙	⊙	⊙	
80	2371601								⊙	⊙	⊙	
81	2223823							⊙	⊙	⊙	⊙	
82	2208330							⊙		⊙	⊙	
83	2223689							⊙		⊙	⊙	⊙
84	2396669								⊙	⊙		⊙
85	2223782								⊙	⊙	⊙	
86	2208391							⊙	⊙	⊙		⊙
87	2396650							⊙		⊙	⊙	
88	2384993							⊙	⊙	⊙		⊙
89	2208390							⊙		⊙		⊙
90	2350073							⊙		⊙		⊙
91	2208585							⊙		⊙	⊙	
92	2371589							⊙		⊙	⊙	
93	2208575								⊙	⊙	⊙	

Novel		Authoring		Animated				Still			Type	
	ACM ID	Languages	Tools	Walkthrough (S)	Walkthrough (P)	Animation	Video	Sketches	Pictures	Text	Motion	Touch
1	2466192							⊙	⊙	⊙	⊙	⊙
2	2481330							⊙	⊙	⊙	⊙	⊙
3	2466457				⊙			⊙		⊙	⊙	⊙
4	2470704								⊙	⊙		⊙
5	2468587							⊙		⊙	⊙	
6	2468592							⊙	⊙	⊙		⊙
7	2468602			⊙				⊙		⊙	⊙	⊙
8	2468636								⊙	⊙		
9	2468645							⊙	⊙	⊙		
10	2468525							⊙	⊙	⊙		⊙
11	2491368				⊙				⊙	⊙	⊙	
12	2491593							⊙		⊙	⊙	⊙
13	2451207						⊙		⊙	⊙		
14	2449435				⊙				⊙	⊙	⊙	
15	2459238								⊙	⊙	⊙	⊙
16	2459240							⊙		⊙	⊙	
17	2447571								⊙	⊙	⊙	
18	2460654				⊙			⊙	⊙	⊙		⊙
19	2460668								⊙	⊙	⊙	
20	2460661							⊙		⊙	⊙	
21	2318005						⊙	⊙	⊙	⊙	⊙	⊙
22	2168578				⊙			⊙	⊙	⊙	⊙	
23	2305306							⊙		⊙	⊙	⊙
24	2254585									⊙	⊙	
25	2414585							⊙		⊙		⊙
26	2318071								⊙	⊙	⊙	
27	2399149							⊙		⊙	⊙	
28	2254590							⊙	⊙	⊙	⊙	
29	2148195							⊙		⊙	⊙	
30	2384969								⊙	⊙	⊙	
31	2148148						⊙		⊙	⊙	⊙	
32	2148172							⊙		⊙		
33	2223673							⊙		⊙	⊙	⊙
34	2371712						⊙	⊙	⊙	⊙		⊙
35	2407360				⊙			⊙	⊙	⊙	⊙	
36	2396697							⊙		⊙		⊙
37	2208297							⊙		⊙	⊙	⊙
38	2371711						⊙	⊙		⊙		⊙
39	2350098							⊙		⊙	⊙	⊙

* Under ACM classification H.5.2 Information Interfaces and Presentation: User Interfaces—Input devices and strategies
* Years 2012 and 2013 (Until 22.08.2013)
* Novel: New and Novel Interaction Technique Paper
* Use: Gesture based Interaction Paper (using interactions, analysing interaction techniques, application scenarios, etc)

Figure 5.6.: ACM multitouch and motion based interaction papers review - interaction documentation practices and habits analysis for the years 2012 and 2013 (until 22.08.2013)

Figure 5.6 illustrates the complete classification of the analyzed papers based on the previously presented methodology. Papers that satisfy the conditions are distinguished with a coding mark as shown in the figure. The analyzed papers were motion (51%) and touch based (68%) interaction papers (note that a paper may fall into more than one category).

Observation - P1:

Multitouch- and motion-based interactions in publications are mostly documented using text, sketches, and pictures respectively.

As expected, figure 5.6 shows that text descriptions as a medium for documenting interaction techniques are used in all of the papers that we have reviewed. Sketches (59%) come second with a very close match with the designer survey in Figure 5.2. Pictures (53%) come third, slightly lower than in the designer survey.

Moreover, personal walkthrough is reported by 16% (the developer introduces the interaction technique to other developers or users by demonstration). Videos are reported by 11%. This percentage matches the survey's results (Figure 5.2). In research papers, mentioning and linking to video content is usually neither required nor critical for the acceptance of the research paper. Hence, videos related material to the technique are often hidden. The use of animations is reported only once. This matches to a large extend the designer survey results in Figure 5.2. On the other hand, other media types such as onscreen walkthrough are hardly used.

Observation - P2:

Multitouch- and motion-based interactions in publications are never documented using formalized languages or interaction authoring tools.

To our expectations, none of the papers reported or used languages (including notations and formalisms) or interaction authoring tools (including gesture authoring tools). Finally, we found no statistical difference between the two main aforementioned analyzed groups of papers.

5.4. Analyzing motion-based applications markets initiatives

The availability of NUI enabled commodity devices encouraged new types of application market places specialized and build around gesture interactions. In this part of our study, we have reviewed a number of new and established commercial application market places and initiatives for gesture controlled applications. Our goal was to identify possible differences to the research community's documentation practices.

First, we have reviewed the publishing guidelines for the Android Play Store[6] and Apple App Store[7] as two of the most popular multitouch- and motion-based application market places. Principally, those market places allow promoting materials limited to graphics, screenshots, videos, and text descriptions. The guidelines are purely concerned about formatting, corporate identity, copyrights, and promotional issues. Nonetheless, the content is left open to developers with the emphasis that it should reflect the application's look and feel, the way it's used, and its main features. Currently, both market places don't offer any guidelines on describing multitouch- and motion-based gestures and don't support or allow formalisms or unified ways (e.g., templates and gesture languages) to describe the required interactions. Likewise, more dedicated new market places initiatives for motion-based interactions including MAGECA[8] (market place for gesture controlled apps and games), OpenNI community portal[9] (market place for 3D sensing middleware libraries and applications based on the OpenNI framework), and Motionfair[10] (market place for gesture controlled apps and games) lack adequate documentation of movements for interactions and only rely on limited set of generalized promotion materials.

We have analyzed 35 motion related products out of 72 unique advertised projects in the three market initiative stores (i.e., MAGECA, OpenNI, and Motionfair). Our filtering criteria excluded all none touch or motion gesture products and duplicated products in multiple market places. Our classification and categorization of those products were based on the market's associated material and descriptions for each project, in addition to any additional linked resources, e.g., dedicated website. The products were split into three main groups: (1) applications; (2) games; and (3) systems and utilities. Each product was coded according to five aspects: *Availability* - the products access types including free products, commercial products, and

[6] https://play.google.com/store, latest access on 20.04.2014.
[7] https://itunes.apple.com/en/genre/ios/id36?mt=8, latest access on 20.04.2014.
[8] http://www.mageca.com/, latest access on 20.04.2014.
[9] http://www.openni.org/, latest access on 20.04.2014.
[10] http://www.motionfair.com/, latest access on 20.04.20134.

showcases; the remaining categories (*Type*, *Still*, *Animated*, and *Authoring*) are identical to those we have described in the previous step.

> Observation - **M1**:
>
> NUIs in market places are mostly documented using videos, text, and pictures respectively.

> Observation - **M2**:
>
> NUIs in market places are never documented using formalized languages or interaction authoring tools.

Figure 5.7 presents the analyzed products in the three market places as indicated by our methodology. The vast majority of the current products are motion-based products. 97% of the products in our sample adopted videos for describing the involved interaction gestures, followed by text with 51%. Pictures and sketches were also identified with 29% and 14% respectively. 9% of the products indicated an implemented onscreen walkthrough in the product. None of the products mentioned or used formalized languages or interaction design tools for authoring interactions.

Compared to the survey's results shown in Figure 5.2, the use of text matches perfectly the survey's results. Nevertheless it does not match the design section which reported a use percentage around 85%. Moreover, the percentage of video use is much higher than the results of the survey. This result may be biased due to the overlap between the available dedicated illustration videos and marketing videos (used mostly in the reviewed markets). The use of pictures and sketches is much lower than within the survey, due possibly to product developers' focus on marketing materials and paying less attention to the actual use and instructional materials of the product.

A proper description of multitouch- and motion-based applications should reflect just enough information for the user in order to take an informed decision to understand the application's offered functionality (e.g., map navigation), the physical abilities and movements required (e.g., interaction with one or two hands), and whether the application matches the user's personal style and preferences (e.g., jumping interaction with one foot does not suit all elderly users). We believe that a more extensive and separate investigation is required for a complete and more representative analysis.

			Availability			Type		Still			Animated			Authoring
		Product Name	Free	Commercial	Showcase	Touch	Motion	Text	pictures	Scetches	Video	Animation	Walkthrough	Language
Applications	1	ExpressionMouse	⊙				⊙	⊙			⊙			
	2	ExpressionMouse PRO		⊙			⊙	⊙			⊙			
	3	NatuLearn Swiper	⊙				⊙				⊙			
	4	Enable Viacam	⊙				⊙	⊙						
	5	KineSis	⊙				⊙	⊙	⊙		⊙			
	6	C-Blues Orchestra		⊙			⊙	⊙	⊙		⊙			
	7	Feel the Moon Gravity	⊙				⊙				⊙			
	8	Fusion4D					⊙	⊙			⊙			
	9	NatuLearn Imqu	⊙				⊙				⊙			
	10	NatuLearn Quiz	⊙				⊙				⊙			
	11	IShow Move		⊙			⊙				⊙			
	12	TedShop			⊙		⊙				⊙			
	13	Interactive Video			⊙		⊙				⊙			
	14	LikeShow		⊙			⊙				⊙			
	15	ZD MotionWall		⊙							⊙			
Games	16	Kinect PONG	⊙				⊙	⊙			⊙			
	17	Kinect TOUCH 2.0	⊙				⊙	⊙	⊙		⊙			
	18	Kinect MONSTERS 2.0	⊙				⊙	⊙	⊙		⊙			
	19	Kinect FRUITS 2.0	⊙				⊙	⊙	⊙		⊙			
	20	Falling Objects	⊙				⊙				⊙			
	21	SkiRanger	⊙				⊙	⊙		⊙	⊙			
	22	BioGaming			⊙		⊙							
	23	KinectoTherapy	⊙				⊙	⊙		⊙	⊙			
Systems and Utilities	24	GestureKey		⊙		⊙		⊙		⊙	⊙		⊙	
	25	Kinetic Space	⊙				⊙				⊙			
	26	HandKinetics			⊙		⊙				⊙			
	27	Map Dive User			⊙		⊙	⊙	⊙		⊙		⊙	
	28	Fitnect			⊙		⊙		⊙		⊙		⊙	
	29	YumeWe			⊙		⊙				⊙			
	30	Finger-Precise		⊙			⊙	⊙	⊙	⊙	⊙			
	31	Po-motion		⊙			⊙				⊙			
	32	Ayotle Anytouch			⊙		⊙		⊙		⊙			
	33	ZiiCON-ad		⊙			⊙	⊙			⊙			
	34	ViiMotion		⊙			⊙		⊙		⊙			
	35	g-speak™ SDK		⊙			⊙	⊙			⊙			

Figure 5.7.: Motion-based application market review - interaction documentation practices and habits (22.08.2013)

5.5. Discussion

Based on the above discussed results and observations, we have identified a number of interesting aspects and issues, which will be presented and discussed in this section:

5.5.1. Interaction documentation: luxury or necessity?

So far, our observations (D2, D3) unveil that documentation is generally an underestimated or ignored issue. The HCI researchers' effort is focused on aspects such as novelty, usability, and impact. A long term documentation of interaction techniques may be considered a luxury rather than a necessity. Documentation in fields that relay heavily on movements such as dance is traditionally challenged by the limited economics, time, and resources [158]. Similarly, those reasons may also apply in the interaction context as well.

The bulk of research in this field strives to create interaction techniques that are easy to learn, natural, and self-explaining, hence documentation may be seen unnecessary. This correlates to one of the most important observations regarding learning habits, which indicates that users rely greatly on try and error to learn interactive techniques (U6). This can be also the result of the limited range and simplicity of interaction features currently available in the users' commodity devices (e.g., swipe, shake, and pan).

There is a strong evidence that learning and memorizing interaction techniques will become more complex due to the vast growth of multitouch- and motion-based interactions in terms of, but not limited to, the number of interactions proposed, the increasing complexity of interaction techniques, expanding diversity of interaction types, involved body parts, involved actions, and runtime ensembles of interaction techniques [51][56][122][10]. This clearly advocates a reference documentation of interaction techniques as a necessity and a great aid to the users.

Moreover, our investigation reveals that the users' learning habits are not currently optimized, and a mismatch between current and preferred learning practices exists (U7). This calls for actions to be taken by the research community to bridge this gap and to better consider user needs. We believe that documentation should be made easily accessible to match the user expectation in multiple forms and methods (for instance, providing a guided try and error approach).

The shift towards future ambient spaces imposes new requirements, and challenges the current practices. Pruvost et al. [137] noted that interaction environments are becoming increasingly heterogeneous and dynamic, hence they are no longer static and closed; the interaction context is becoming increasingly more complex; and, increasing adaptability is required for sustainable utility and usability. Recent approaches for ad-hoc composition of multiple interaction techniques at runtime, as proposed in this dissertation, certainly add more challenges to the current practices, e.g., dynamically created documentation in-situ. Interactions are currently ego-centric and designed in isolations, so is the documentation. Such isolation implies a complete absence of information about the interaction's behaviour as part of an ensemble in a dynamically changing eco-system.

Finally, good record-keeping of interactions is very important for evaluating interaction's required physical movements, reconstructing interactions correctly from an interaction heritage, and accessing interaction techniques more reliably by the community being research or commercially oriented.

5.5.2. Documentation styles: Freeform or formalized?

Our observations indicate that the documentation style is an open issue in the field of interaction design and development. The results show that various media types, such as textual records and visual records, are used for documentation differently among designers. At the same time, formalized languages are hardly applied (D3, D6, P2). This can be related to the complexity of language learning, the complexity of describing movements, and the lack of formalized languages and notations of generic motion (D4 that matches Kahol et al. [84] findings).

Formalized languages have clear benefits to depict the interaction technique without endangering its originality and vital aspects. Currently applied media such as text, pictures, sketches, and videos may lead easily to losing parts of the movements, overly complicated descriptions, losing timing information, etc. In fact, according to Navarre et al. [122], formal interfaces description languages support interaction at the development (e.g., prototyping) as well as the operation phase, while conventional empirical or semiformal techniques lack to provide adequate and sufficient insights about the interaction (e.g., comparing two design options with respect to the reliability of the human-system cooperation). According to Kahol et al. [84], having such languages and notations features three main qualities: facilitate teaching and learning of movement styles, permit the writing of universally-understood scores of movement, and provide a universal language to communicate movements. Similar to Navarre et al. [122], we argue that lacking adequate and formalized documentation leads inevitably to increase the gap between the design and (commercial) deployment of developed interaction techniques.

5.5.3. The importance of documentation

Designers recognize the importance of documentation for the users' experience, the acceptance of NUI techniques, and correctness of use. The most important use of documentation is for sharing NUI techniques (D7). Sharing is particularly important for different purposes such as communicating NUI to other peer designers, improving NUI functionality by other designers, adopting NUI techniques in various interactive eco-systems, and reaching user audience. Even though designers recognized these important roles, their documentation practices appear generally ignorant to this importance.

5.5.4. Documentation and learning challenges in future ambient spaces

From the NUI user point of view, Interaction Ensembles offer a highly adaptive interaction environment to her physical abilities, but at the same time challenges her cognitive abilities to learn and understand the composed ensemble. With increasing interaction possibilities and combinations, currently applied user learning strategies, such as "Try and Error", become less effective and fail to scale for complex interaction scenarios.

From the designer point of view, typical NUI documentation practices are greatly challenged by such a scenario. The current documentation practices and strategies are not adequate, and fail to meet the challenge of dynamically created documentation for interaction ensembles. Designing isolated interactions implies a complete absence of information about the interaction's behaviour as part of an ensemble in a dynamically changing eco-system. Moreover, the creation of documentation for an interaction composition from multiple *heterogeneous multimodal documentation* resources is another challenging aspect.

5.6. Conclusion

In this chapter, we have presented an investigation on the applied practices and habits to document and share developed interaction techniques. The analysis included: (1) an online exploratory survey on documenting Natural User Interfaces (NUI) answered by 64 designer and 267 end user; (2) analyzing a sample of 93 recently ACM published multitouch and motion-based interaction papers; and (3) analyzing three new motion-based applications market initiatives.

In this chapter, we substantiated the following main contributions and findings:

- We have presented a number of observations regarding the NUI designers' most commonly applied documentation choices, most importantly, documentation frequency and media type of choice (covered in section 5.2.1).

- We have presented a number of observations regarding the NUI users' learning habits and preferences, most importantly, learning methods and used documentation media types (covered primarily in section 5.2.2).

- We have discussed the designers' use of availability of adequate documentation tools, documentation standards, and the regularity of documentation habits (covered in sections 5.2.1). This issue will be further investigated in section 6.2.

- We have reviewed some related application market stores and their general guidelines for publishing multitouch- and motion-based applications (covered in section 5.4).

- We have discussed the match and mismatch between the NUI designers' documentation practices and users' preferred learning practices (covered in sections 5.2.1 and 5.2.2).

This study reveals that good documentation practices are rare and largely compromised due to the lack of adequate documentation tools, absence of documentation standards, and irregularity of documentation habits. Moreover, the investigation highlights the impact of documentation practices on the users' learning practices, especially by exposing the existing mismatch gaps between preferred and available documentation methods and materials for users. Hence, the creation of a collective long lasting *interaction heritage* remains unachievable and optimal user learning habits remain unsatisfied and weakly considered.

Finally the investigation aims to trigger a community-scale discussion to consider documentation as an important design measure for successful preservation, dissemination, and sharing of interaction techniques.

6
Documenting Natural User Interfaces

6.1. Introduction

Parts of this chapter appear in [10], [7], and [11].

This chapter covers a detailed overview on movements and abilities documentation for NUI and presents the main design and implementation choices considered in our work. This chapter is structured in four sections. The chapter starts with an introduction (section 6.1) that includes the motivation and importance behind this subject (6.1.1), movement documentation in practice (6.1.2), and movement and interactions (6.1.3). Next, section 6.2 presents an analysis of existing documentation-related tools and languages for interactions. Section 6.3 presents our investigation towards documenting Kinetic Interactions. The implementation of movement profiles is discussed and presented in section 6.4, this includes a lengthly discussion about Labanotation as one of the most comprehensive systems for documenting movements (6.4.1), the developed machine readable XML representation of Labnotation (6.4.2), and examples and showcases (6.4.3). Moreover, the chapter discusses the implementation of the physical ability profile in section 6.5. Next, the chapter presents a tool specifically implemented for authoring movements for NI called the Interaction Editor (6.6). Finally the conclusions are presented in section 6.7.

6.1.1. Motivation

Decimation and sharing strategies for NUI are primary cornerstones for the concept of Interaction Ensembles. In this chapter, we argue that documentation languages and strategies are very important for the realization of a dynamic and adaptive interactive eco-system. Our approach adopts body movements as the central focal point in NUI. This has lead to exploring the main methods used to preserve interactions, especially focusing on the movement qualities and physical abilities of the human body.

This chapter strives to investigate the following key questions:

- What are the main existing documentation approaches and tools for NUI designers?

- What are the main challenges for NUI documentation in ambient spaces?
- How NUI can be effectively documented?
- What are the main relevant documentation qualities for NUI?

6.1.2. Movement documentation in practice

Movements documentation is a very relevant and generally a very unresting problem for many fields such as dance choreography, movement rehabilitation, motion recognition and analysis, and human movement simulation. Nevertheless, it remains a challenging task due to the lack of formalized languages and notations of generic motion. According to Kahol et al. [84], having such languages and notations features three main qualities: facilitate teaching and learning of movement styles, permit the writing of universally-understood scores of movement, and provide a universal language to communicate movements.

Generally, the process of documenting movement should preserve a number of qualities including movement sequences and timing, flexible level of abstraction, ambiguity elimination, and the body parts involved in the movement. Alaoui et al. [3] classified movement qualities definitions into two categories, influenced by either body expression or motor theory.

Movement qualities as body expression have been the focus of dance-related studies, especially by the Laban Movement Analysis (LMA) research lead by Rudolf Laban. LMA defines movement in terms of four dimensions [77]: (1) Body - presenting what is moving (i.e., body, body part, object, etc); (2) Space - presenting where the movement takes place; (3) Shape - presenting the body changing its posture and shape during the movement; and, (4) Effort - presenting the characteristics of movement in respect to intentions, which has direct impact on the movement dynamics (i.e., intentions change movement degree, strength, timing, etc.).

On the other side, the movement theory, used often by psychologists, is mainly concerned about the movement execution with respect to time and space [3], but also covers the link between emotions and movement qualities as in [173].

Although Labanotation is used in LMA, it is still a separate system and regulated separately.

One of the most notable systems for physical movement recording and analysis is called Labanotation (Kinetography) [77][84]. Although it started out as a means to capture dance, Labanotation is universal enough to capture any kind of movement. In addition to Labanotation, different successful systems were introduced such as Benesh Movement Notation (Figure 6.1) to record ballet and Eshkol-Wachman Movement Notation

(Figure 6.2) that employs a spherical system of reference to describe the paths and directions of movements and limbs [77].

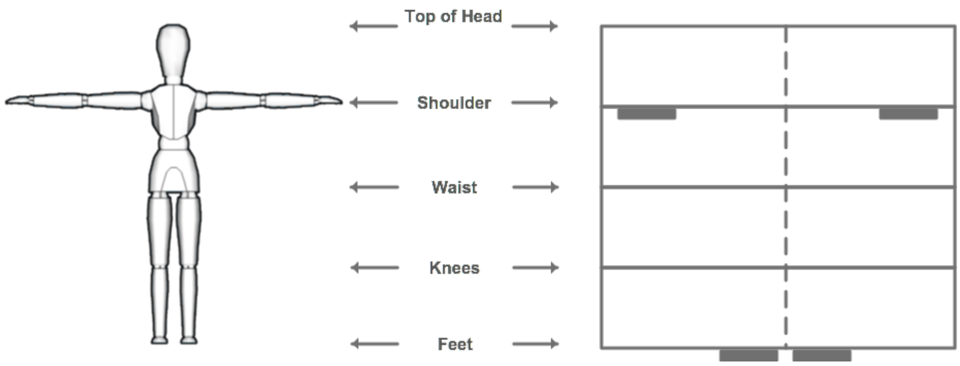


Figure 6.1.: A simple illustration of the Benesh notation

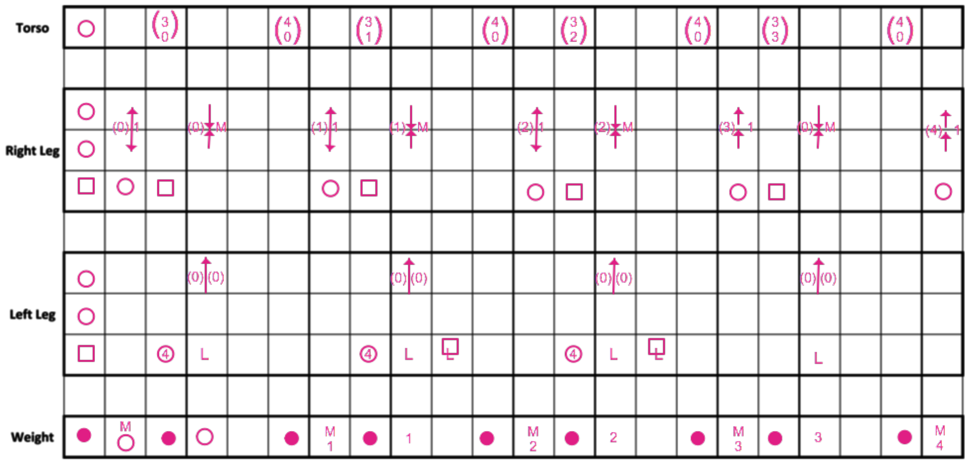


Figure 6.2.: A simple illustration of the Eshkol-Wachman Movement Notation

Labanotation is adopted for our approach due to its flexible expressive power and holistic power to capture movements in terms of movement structural description, analysis of patterns (shapes), and qualities of movement (efforts). In our approach, those components are essential ingredients to document movements for NUI. Additionally, Labanotation fosters great flexibility that empowers designers to document all or any part of movements as required.

A handful of projects were reported to compose and visualize dance choreography and animation of human movement such as Life Forms, Virtual Ballet Dancer [104], LabanWriter[1], LabanEditor [98], Tiniklink and

[1]http://dance.osu.edu/labanwriter, latest access on 15.03.2014.

MovementXML [70]. Unfortunately those projects are not suitable for our purposes because they either are using none machine readable formats, not available for open source development, or are not published. Additionally, these projects aimed at a complete representation of dance rather than interactions, hence provided a more complex set of notations that potentially increases the complexity to adopt those projects for designing Kinetic Interactions.

In computer systems, the bulk of previous works on Labanotation was targeted at digital representations of human movement. Labanotation was used in motion recognition computer systems to extract movement emotive qualities as in [27], designing affective input [54], and gestural semantics of caress [149]. Opposite to our approach, the structural aspects of movements were not considered as the core point of these projects.

In computer generated graphics, Labanotation is used to increase the naturalness and expressiveness of simulated movements as in the EMOTE model [36]. There have been yet very few attempts to automatically perform Labanotation-based movement analysis and recording. Hachimura et al. [66] were the first to develop such a system based on 3D motion capture data. They were able to record the orientation of the various body segments (as well as the gesture boundaries) in a textual description named Labanotation Data (LND). Moreover, a vision-based 3D interface was proposed by Woo et al. [185] based on extracting movement qualities from LMA. Using digital media to document contemporary dance was explored in the Movement Knowledge project (IMK) [55][3]. The IMK project aimed to unveil the potential for digital media to document dance by an interdisciplinary research including linguistics, dance notations, motion capture, digital media, and recognition algorithms and glossary. Despite the aforementioned effort, documenting movements is still a challenging and unresting problem and more investigations are required towards robust 3D motion capture technologies, standard movement description languages and notations, and movement authoring tools and editors.

6.1.3. Movement documentation for interactions

A background review on movements and interaction design is presented in section 2.2.3

The literature lists a few projects that adopted Labanotation for interactions. One of the earliest work to investigate the potential use of Labanotation for designing and analyzing interactions was reported by Loke et al. [104]. In their work, they have presented an analysis of people's movements when playing two computer games, which utilize players' free body movements as input sensed by a basic computer vision. They have named a number of ways to describe movements, ranging from the mechanics of the moving body in space and time, the expressive qualities of movement, the paths of movement, the rhythm and timing, and the moving body involved in acts of perception as part of human action and activity [104].

Kahol et al. [84] suggested modeling motion gestures as a sequence of activity events in body segments and joints, in order to provide an intuitive method to understand the creation and performance of a gesture. A motion sequence, once captured, can be annotated by several different choreographers, based on their own interpretations and styles. Their work was proposed as method for developing a common language of motion.

According to the studies cited in this section, existing efforts neither target NI documentation per se nor provide a general methodology to tackle this highly important problem specifically. This puts the issue of interaction documentation as a clear challenging aspect for the core concept (Interaction Ensembles) suggested by this dissertation. The rest of this chapter will tackle this issue in more detail.

6.2. Documentation-related tools and languages

The dissertation's quest to identify interaction documentation tools has led to investigate tools, approaches, and languages for authoring interactions, prototyping interactions, gesture visualization tools, and gesture databases. This is the fourth area of our investigation, covered in Chapter 5. It was mainly targeted to identify the use of movement documentation languages and tools. Our initial strategy was intended to collect a list of the most used and popular tools mentioned by the reviewed scientific papers. Unexpectedly, the analysis did not result into a satisfactory set of tools. None of the papers reported or discussed any used documentation-related tools or languages, shown in Figure 5.6 (appeared in section 5.3 "Scientific publications"). Hence, we based the analysis on our own literature review about the topic. The reviewed tools set is neither complete nor compressive. Nonetheless, we consider this step as a starting point for the discussion, specially if combined with the results from the other investigation steps resorted in Chapter 5.

Tools for authoring interaction techniques have promised interaction designers and developers easier and more productive designing, prototyping, and development environment. Programming by demonstration is one approach where a set of target interaction behaviours are learned from examples provided by the developer, as in Gesture Coder [106] for multi-touch gestures and ANID (Authoring Natural Interactions by Demonstration) for motion based interactions [114]. In Gesture Coder, additional to the internal state-machine used for detecting the gesture events, the only form of documentation available is through gesture view. Gesture view provides designers with a video capture of touch points on the surface. But this presentation is not rich in terms of amiable clues about gestures.

For instance, information about which figure or hand should perform the gesture is not visible. In the case of the ANID project, a visual tool called "OscMotivator" is created to demonstrate gestures either through temporal curves or by an animation. Although it was created for demonstration purposes, we consider this effort as a form of documentation.

Declaration-based approaches adopt a behaviour specification using high-level languages or representations. The Midas framework [150] aimed at gesture reusability and Proton++ framework [95] aimed at prototyping multitouch gestures visually using regular expressions. Although the Midas framework allows to describe gestures to be later used in different applications, it does not provide a way to describe how those gestures should be performed. Only simple text descriptions are used as an indication, i.e., tap, drag, rotate, zoom, etc. On the other hand, Proton++ framework allows designers to describe gestures as regular expressions of touch events symbols. This is one of very few frameworks that can be used for documenting multitouch gestures and the visual notations allow for a standardized way of describing gestures. While this is a good milestone to document interactions, the framework is not applicable for motion gestures and falls short in terms of describing the multitouch gestures fully. It is not possible to convey any information about the involved body part(s) (which hand/finger is involved?), the physical posture state (what hand posture should be maintained?), etc.

Gesture Studio [107] is an available tool for designing and prototyping multitouch interactions, which employs the best of the two approaches mentioned above. It adopts a video-editing metaphor to create multitouch behaviours and allows developers to edit and compose demonstrated gestures. It combines machine generated code and captured gesture clips (analogous to video clips) for prototyping and recalling behaviours. The gesture clips are used to capture the touch and stroke actions of the user on the surface. Nonetheless, the capturing of the hand and finger postures and movements is not supported by Gesture Studio. Moreover, Gesture Studio's support is limited to multitouch gestures.

Hoste [74] presented the first steps to advocate the use of a rule language to derive useful patterns out of the events generated by the multitouch device. The main goal aimed at a domain-specific language supporting spatio-temporal operators. In terms of documentation, this work is still in its initial stages and is considered narrow in focus. It falls short in describing the gestures and how they should be performed. Furthermore, the Gesture Definition Language (GDL) [93] is introduced as part of the TouchToolkit. The language aims at integrating natural and meaningful multitouch interaction without worrying about implementation complexities. Yet, GDL does support some multitouch gestures and does not include any support for motion gestures.

On the motion gesture side, the Multiple Action Gesture Interface Creation (MAGIC) [13] was introduced to support motion gesture developers during the interaction design lifecycle, consisting mainly of creating, testing, and exporting gestures to Everyday Gesture Library. The authors identified four main requirements for such a tool, namely, support for expert and non-expert usage; encourage iteration; retrospection; and, further testing. The tool allows developers to examine previously created content and review the actions taken by plotting recorded gestures and recorded video of the designer performing the gesture. We believe that MAGIC is one of few projects that use gesture capture and video recording as a documentation for motion based interactions. Nonetheless, MAGIC is merely targeted at designers and only supports the design cycles of the interaction development. Neither deployment cycles nor users are supported by MAGIC. Available documentation materials, i.e., captured videos, are not exposed to users as a learning and documentation material.

Gesture visualization tools are another area of interest for NUI documentation. Visualizations can also be used for gesture interfaces as a feedback and as an animated tutor for users. Out of a very few research projects in this area, Kallio et al. [85] proposed a visualization of hand gestures and their main goal was to investigate the visualization impact on gesture control performance. Their study indicated that visualizations can be used to build a mental picture of gestures and increase correct associations and executions of gestures. Likewise, the ActionCube project [102] aimed at a mobile-based tangible interaction tutorial to serve as a gesture interaction tutor for a new user, while offering an entertaining user experience. Additionally, EventHurdle [94] is a visual gesture authoring tool to support designers explorative prototyping for motion based and multitouch gestures. EventHurdle presented three important features: an interaction workspace visualizing various types of sensor data in a unified way; extensible definition scheme for gestures using a graphical markup language (that can describe spatial movements); and, finally an automated code generation process.

Chen et al. [34] discussed the need for motion gesture databases. They have proposed the 6DMG motion database that contains comprehensive motion data, including the position, orientation, acceleration, and angular speed for a set of common motion gestures performed by different users. Although expected, the database does not provide, nevertheless, well defined documentation material, i.e., information and language descriptions, on the motion gestures. Moreover, the execution aspects of gestures, i.e., movement components, posture, etc., are left out of the database. Based on this review, we have identified the following four observations:

Observation - T1:

None of the available tools or approaches are widely adopted by NUI designers.

Observation - T2:

There are no dedicated available tools for NUI documentation. This is because most tools are either experimental prototypes or limited research efforts, which are also not specifically aimed at NUI documentation per se.

Observation - T3:

Most tools lack the support for end NUI users.

Observation - T4:

Current tools lack the support of anthropometric body movements and postures descriptions as part of the interaction description.

The observations T2 reveals that no dedicated NUI documentation tools were found, instead reported tools merely cover partial aspects of NUI documentation. Furthermore, only few tools are currently available for designers to use, such as Gesture Studio [107], Proton++ [95], and 6GMG [34]. Other reported tools are either proof-of-concept prototypes to ease the process of interaction authoring or simply unavailable.

The observations T3 clearly indicates that most of the projects identified in the scope of this chapter aim to ease the process of prototyping and authoring of interactions at design cycles. Only a few prototypes explicitly aim at end users and support deployment cycles of NUI such as Kallio et al. [85], ActionCube project [102], and EventHurdle [94].

According to observation T4, NUIs are inherently susceptible to a wide range of physical impairments and diverse interpretation of required body movements for an interaction. Hence, exposing adequate information about anthropometric body movements involved for an interaction is highly recommendable for context-driven adaptation and for increasing the reproducibility of interactions, which cannot be adequately satisfied otherwise. As discussed earlier, the available tools as well as gesture databases lack the support for such information.

The interaction documentation process should go beyond the available tools. Preserving the constructing physical movements of the interaction should be a very integral part of any documentation. Such information guarantees to protect and preserve a number of interaction qualities including movement sequences and timing, flexible level of abstraction, ambiguity elimination, and the important parts of the movements [7]. Finally, the lack of a formalized language and notation of generic motion remains a challenging task for the HCI community [84][7].

6.3. Towards documenting NI techniques

Documenting interactions is an essential cornerstone for the learnability and utility of interaction techniques by users, designers, and developers. HCI researchers tend to preserve and describe newly developed NI techniques using one of four main practices, namely; direct personal transmissions, written textual records, still visual records (e.g., images, sketches, drawings), and animated visual records (e.g., videos and animations).

An overview on used documentation practices is covered in Chapter 5

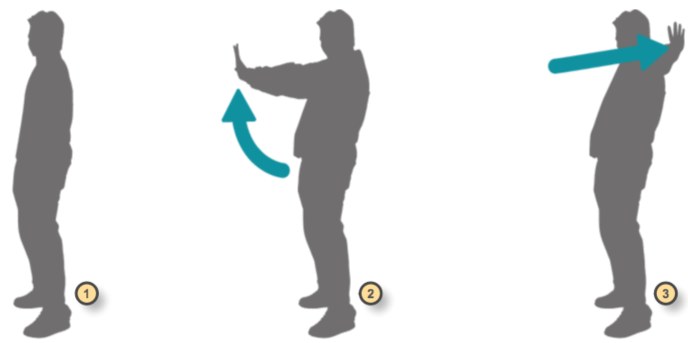
Nevertheless, the aforementioned methods suffer from different drawbacks [10], which may impact the documentation quality, e.g., textual records are often too ambiguous, inaccurate, or too complex to comprehend; still visual records fail to convey timing and movement dynamics (e.g., parallel movements may be obscured by each other); animated visual records are affected greatly by the capturing quality, lighting conditions, and filming angle; and inability to utilize different medium to convey the movement (e.g., movements presented in a sketch are only provided in that form).

In fact, describing interactions in ambient spaces is a much more challenging task because of:

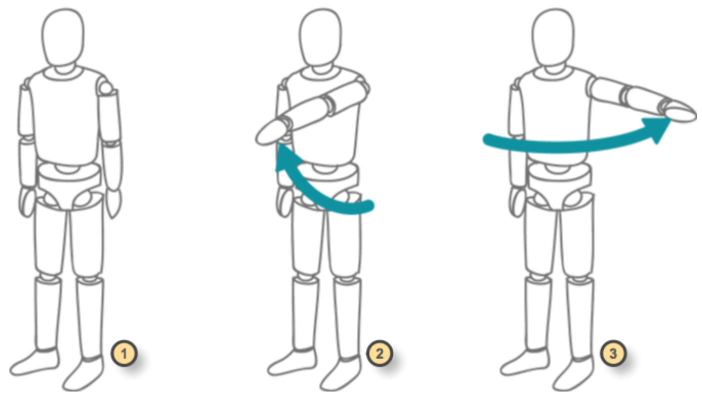
- the heterogeneity of users' needs and abilities,
- heterogeneity of environment context, and
- media renderers availability.

To demonstrate one of the many issues regarding current documentation practices, Figure 6.3 shows two different drawings of the same interaction technique. The technique presented in the drawings is a simple arm swiping gesture. The gesture requires the user position the left arm to the front parallel position to the ground (as a starting position), and move it to the right side to do a right swipe or to the left to do a left swipe (for interaction). The two drawings depict the interaction differently using different drawing styles, angles, ways to depict sequencing, etc. Both drawings can be easily differently interpreted by users as well as peer-designers causing great variations in interaction understanding and execution. For instance, different drawing angles makes it very hard to recognize ether the hand is (3) is straight, up, or down in Figure 6.3(a) and Figure 6.3(b). Moreover, it is hard to identify the most important aspects of the movements. Figure 6.3(a) may be easily interpreted as a torso movement slightly to the back is required, while the focus of this illustration is only on the movements of the left arm.

Our thanks go to Michal Janiszewski for drawing Figure 6.3 specifically for our work.



(a) Designer drawing 1



(b) Designer drawing 2

Figure 6.3.: Documenting an arm swipe interaction by drawing (Illustration examples)

This dissertation argues that documentation is not only essential for sharing interactions, but also to aid the process of learning new interactions and recalling known interactions, which often are neither natural nor easy to learn and remember [126]. Even the simplest gestures, as simple as a head shaking gesture, may cause a great learning and recalling burden to the user, especially when considering various cultural environments. Moreover, Norman stressed the fact that many developed interaction techniques are novel and unknown to users, hence increasing the expected challenges. In another study on motion-based interactions for elderly [62], it was reported that recalling gestures was too challenging for the study participants.

Hence, we argue that the current documentation practices are not fully suitable for ambient spaces because:

- the lack of standardized and agreed on documentation methods,
- current practices are too static and fixed to a particular media type, which may easily limit the target users of the interaction technique,
- current methods such as direct personal transmissions fail to scale with a massive user population, and
- current practices fail to clearly reveal the required physical abilities to perform the interactions.

Extending the previously mentioned list of language qualities by Kahol et al. [84], successful documentation of natural interactions in ambient spaces should ensure:

- a standardized machine readable and parsable language,
- generation of documentation learning and presentation material (e.g., visual records, audio records, etc.) based on the context of the user and his environment,
- methods for observing users interactions and provide suitable feedback and adaptation, and
- to depict clearly the required interaction movements and physical abilities.

Our approach to document kinetic interactions is based on three main components: Interaction Profile, Movement Profile, and Physical Ability Profile. The combination of those three profiles is one of the essential driving wheels in the NI matching and decision algorithms in the process of creating NI ensembles based on the physical context and physical abilities required by interactions. The following sections (especially sections 6.4 and 6.5) discuss in details this dissertation approach to tackle and implement the documentation aspects for ambient spaces.

Interaction profile first appears as part of our conceptual model in section 3.7.2 and is discussed as part of Chapter 4 (section 4.5.3)

6.4. Movement profile

For successful transmission (sharing) and preservation (description) of NI techniques, recording and analyzing physical movement methods should be applied. This dissertation fosters the use of Labanotation (Kinetography) as a system for documenting physical movements required by NI and represented as a movement profile. Principally, the most important quality required according to our approach is to put body movements as the central focal point in NUI. Labanotation satisfies this quality by providing such an adequate language and vocabulary [104]

Body movements as the central focal point in NUI are discussed in section 2.2.3

This profile is based on a subset of Labanotation (i.e., structural aspects of movement). This part of Labanotation provides an extensible and full description of movements including the body (and its parts) involved in the movement, space information (movement direction, level, distance to other body parts or objects, and degree of motion), timing information (duration and metering), and the dynamix of movement (for instance the quality of movement such as smooth, heavy, and strong) [104]. We have applied this to capture the movement aspects of the interaction. Our flexible modular approach will allow future inclusion of other movement aspects such as movement qualities into this profile.

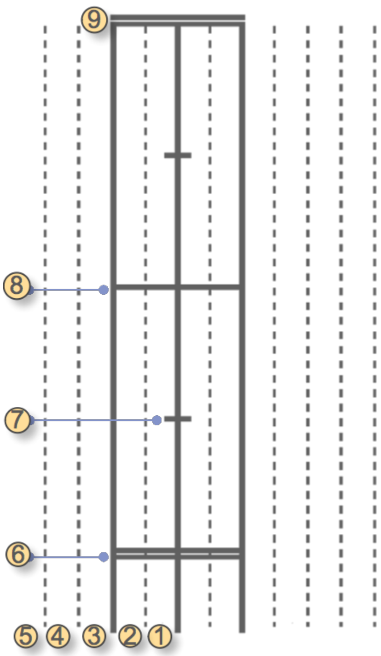
In its current form, Labanotation is a visual notation system. It relies on drawing a vertical score of visual symbols. Therefore, a compliant XML scheme is designed that is both machine and human readable. This scheme allows translating the notation to a machine readable representation of the interaction description. Unlike MovementXML [70], the proposed scheme resamples the standard Labanotation closely for interaction documentation purposes.

6.4.1. Labanotation

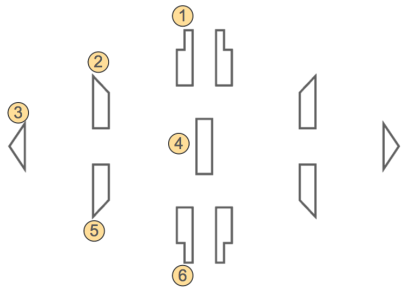
Labanotation is a system of analyzing and recording movement, originally devised by Rudolf Laban in the 1920's. It is then further developed by Hutchinson and others at the Dance Notation Bureau [77]. Labanotation is used in fields traditionally associated with the physical body, such as dance choreography, physical therapy, drama, early childhood development, and athletics.

A complete reference to Labanotation symbols and their meaning can be found in [77]

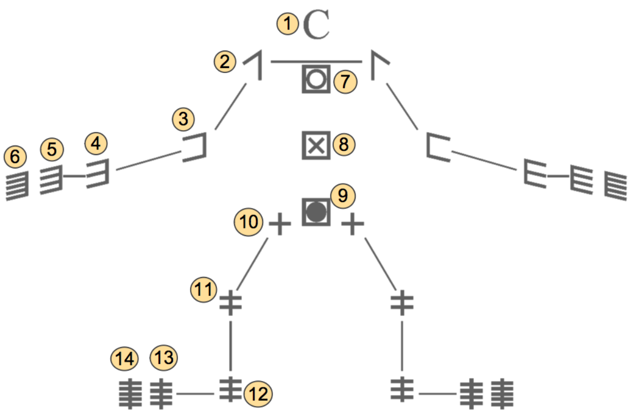
Labanotation comprises a symbolic notation system where symbols for body movements are written on a vertical "body" staff. Figure 6.4 presents a subset of Labanotation symbols used to present and document various movement qualities. Figure 6.4(a) illustrates the Labanotation staff. The staff is used as the layout for all involved movements. Each column, from inside out, presents a different body part. Column (1) presents the support



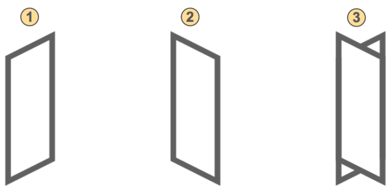
(a) Staff



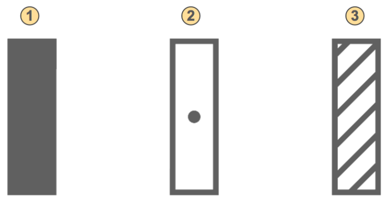
(b) Direction signs



(c) Body parts signs



(d) Turns signs



(e) Movement level signs

Figure 6.4.: Labanotation visual notations (subset)

(i.e., the distribution of body weight on the ground). Columns (2) to (4) present leg, body, and arm movements respectively. Column (5) and additional columns can be defined by the designers as required. The most right column is defined for head movements. The designer is still able to change this order as required by redefining any columns except (1) and (2). The staff is split into different sections. The symbols before the double lines, indicated by (6), present the start position. Moreover, the movements components appear after the position lines in terms of measures (horizontal lines as in (8)) and beats (horizontal short lines as in (7)). The measures and beats define the timing of the movements. The right side and the left sides of the staff correspond to the two sides of the body involved. For instance, arm movements drawn in the right side of the staff present right arm movements.

Figure 6.4(b) presents the direction symbols. Those symbols present the direction of the movements required. The symbols on the right and left sides are for the body parts on the right and left sides respectively. The symbols (1) and (6) indicate forward and backward movements respectively. The symbol (3) indicates a movement to the left and symbol (2) indicates a forward left diagonal movement (45-degree). Likewise, symbol (5) indicates backward left diagonal movement. Symbol (4) is known as the place position (i.e., center position).

Figure 6.4(c) presents the body parts symbols. In Labanotation each body part, surface, or join can be visually presented. The symbols (1), (7), (8), and (9) present the head, chest, waist, and pelvis respectively. The symbols (2) to (6) present the shoulders, elbow, wrist, hand, and fingers of the left hand respectively. The same symbols, but mirrored, are used to preset the right side of the body.

Figure 6.4(d) presents turning movements. For instance, the symbol (1) indicates a clockwise rotation and symbol (2) indicates a counter clockwise rotation. Moreover, symbol (3) indicates a rotation with no specified direction. Other symbols are used to indicate the amount of rotation absolute or relative to other body parts or in a specified angle.

Figure 6.4(e) presents the movement level signs. Those signs are used to indicate the level of the movement required (indicated by the shading of the symbol). Completely shaded (as in (1)) indicates a high or up movement, symbol (2) indicates a middle level, and the symbol (3) indicates a lower or down level movement.

Interaction technique examples modeled in Labanotation are presented in section 6.4.3

It is important to note that those symbols are used individually or combined with other symbols to present the required movement. For instance, a direction symbol (3) darkly shaded as in the level symbol (1) would indicate a movement to the left high side. This ability to combine symbols foster the strength of this notation but at the same time increases the reading complexity of a Labanotation score, especially with large number of symbols and many parallel movements. An extensive coverage of Labanotation notation and symbols is out of the scope of this dissertation.

Labanotation as a recording system for NI movements is very useful and has a number of relevant features such as it:

- is an extensive and flexible notation system,
- is easy to read and write (once familiarity is gained with the notation),
- is very logical and systematic,
- specifies movements from very simple and high level description to very specific movement description,
- has a great expressive power due to it's comprehensive symbol set, and
- enables choices for designers, about what they represent as significant and relevant aspects of movement.

Even though this system is very relevant to HCI research, only few research projects have demonstrated the use of this system to describe interaction techniques such as [104]. This can be the result of many reasons including but not limited to the researcher's lack of familiarity with reading and writing labanotation, the lack of tools for editing labanotation for interaction design, and limited recognition of the importance of documenting and sharing NUI techniques.

In STAGE, we do not only use Labanotation, but we exceed and extend the adoption of Labanotation in NUI design with anthropometric and physical ability profiles, which is very important for adapting to the user's physical context in action.

The richness of the current Labanotation model serves wide range of purposes. At the same time, it requires enormous learning effort and results into an arbitrary complex notation to read. While preserving the extensibility and richness of the notation, we have opted for a subset of the notation to reduce its complexity and simplify its readability as introduced in the beginning of this section (also elaborated in section 6.4.2). To this end, the Labanotation subset models interaction sequences to include different parallel and sequential movements of body parts governed by the Labanotation score, which insure accurate time and sequencing of actions.

Labanotation as a graphical language is very powerful for human readers. It is nevertheless not readable by machines as to our knowledge there is no published research or standards on machine-readable representations for Labanotation adapted by the community. MovementXML was presented in a master thesis [70] but it was neither dedicated to natural interaction techniques nor was the project continued. Therefore, part of our current work is to create XML representation for Labanotation in order to be able to adapt the system in our proposed approach.

6.4.2. Labanotation XML scheme

The first XML scheme draft (v 0.9.0) was implemented by Jan Gröschner in his Bachelor's thesis [63] under our close supervision.

In STAGE, an XML model of this technique is generated and extended with ability profile information needed to execute the model correctly by users. More importantly, interactions steps and movements become well contained in a movement description entity, which can be parsed by the STAGE system for plugin filtering and selection, as well as NI Ensemble formation.

The modeling of Labanotation is challenging due to the extensibility of the notation, size, and variations of symbols. As mentioned earlier, in the scope of this work, a subset of Labanotation is considered. Nonetheless, the extensibility of this scheme is still possible. The current version of the scheme mainly targets the following elements: direction symbols, pins and contact hooks, space measurement signs, turn symbols, vibration symbols, body hold sign, back-to-normal sign, release-contact sign, path signs, relationship bows, room-direction pins, joint signs, area signs, limb signs, surface signs, a universal object pre-sign, dynamic signs, and accent signs.

Figure 6.5 illustrates an overview over the movement profile XML scheme. The original Labanotation naming is reserved to insure compatibility and readability of the scheme. As shown in the figure, the staff is defined in terms of timing information (measures and timing), the body parts involved (by defining the columns), and movement components are defined in the movements element. The movements element contains a collection of elements to define the individual movements, path, the movement directions, an relationships, and phrasing (connecting individual movements together).

Figure 6.6 illustrates a close overview on the movement element. In this element, a single individual movement is fully described. The information modeled includes placement in the score (defined by the column element), timing information (beats, measures, and execution duration), the body part(s) involved (defined by the preSign), and movement quality such as direction, space, turn, vibration, etc. The number and detailed level of movements modeled depend on the designer. The design should model just enough information for ideal execution of the movement.

Generally, increasing the description details will result into a fine preservation and execution of movements details. At the same time, this inevitably causes a large movement profile that results into an increasing complexity of reading and interpretation. On contrary, reduced details results into a simple movement description that is easy to read and interpret. Nonetheless, this leads to losing the details of movements.

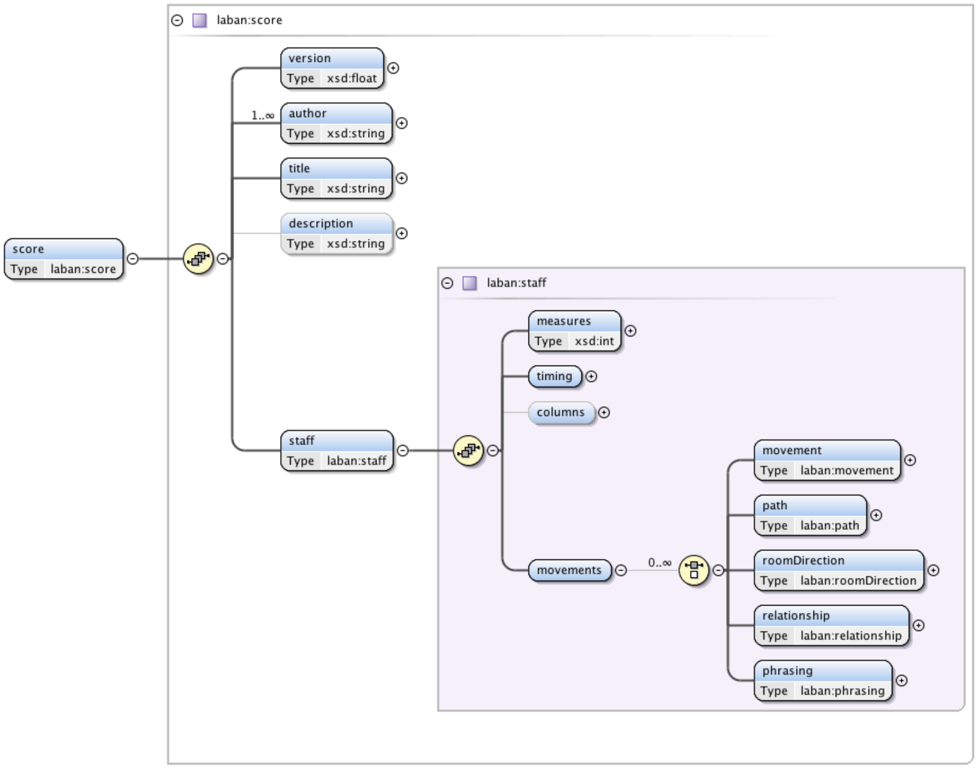


Figure 6.5.: Movement profile scheme - Labanotation score (high-level overview)

In the following section (section 6.4.3), various interaction techniques modeled in Labanotation and their corresponding XML representations are presented.

6.4.3. Examples and showcases

This section illustrates briefly how Labanotation is used in our approach to model various interaction techniques. For readability and reduced complexity reasons, only simple interaction techniques are presented, including the DoubleFlip interaction technique [145], Rahman et al. [139] wrist movements interaction technique, and the 3Gear pinch interactions.

The 3Gear pinch interactions are introduced in section 4.7.1.

Movement profile for the DoubleFlip interaction technique

The DoubleFlip technique is defined as "a unique motion gesture designed as an input delimiter for mobile motion-based interaction" [145]. The authors documented the technique using the following written description: "the user

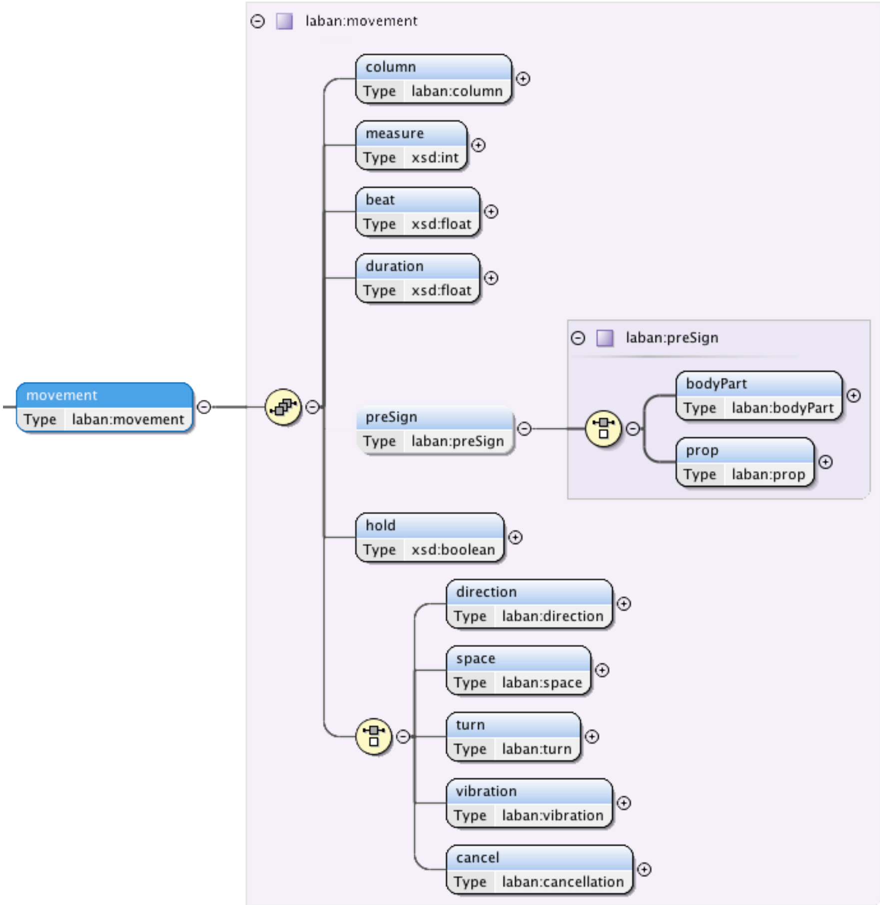


Figure 6.6.: Movement profile scheme - movement element (high-level overview)

holds the phone right-handed, he rotates the phone along its long side so that the phone screen is away and then back". Moreover, they supported the description with an additional sketch as shown in Figure 6.7.

Figure 6.7.: DoubleFlip interaction technique (from [145], used with permissions from the author)

While it was relatively sufficient to relay on text and sketch descriptions for documenting this interaction, it is still relatively hard to explain clearly and insure that the user understands the steps to execute this technique. For example, neither the description nor the sketch clearly illustrates the manner and timing required for this interaction to work. Does the interaction work with very slow hand movement? Is there any break "pause" between the clockwise and counterclockwise movements? Etc. We have modeled the same technique using Labanotation as in Figure 6.8 and Listing 7 (following our XML representation).

The listing is read as follows: (1) the body balance is equal on both legs and (2) stays that way through out the interaction. (3) The starting position of the right is at rest position along side the body and (4) the position of the lower arm to middle front, where arm and lower arm form "L" shape. Both positions are held (5) through out the interaction. Symbols (6) and (7) illustrate the starting position of the hand palm facing up. The wrist performs strong 180-degree counterclockwise rotation (8) and then returns back with palm facing up by a strong 180-degree clockwise rotation (9). Finally (10) the movement is split in terms of timing the described rotation movements.

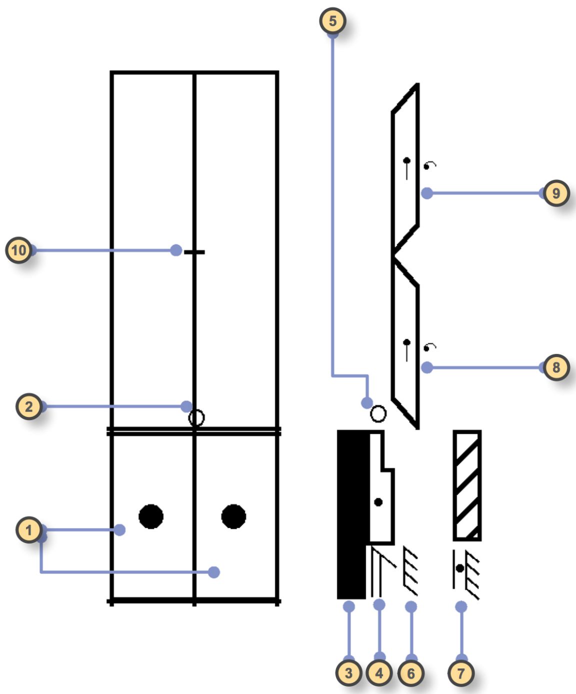


Figure 6.8.: Using Labanotation to document DoubleFlip interaction technique

```xml
<!-- Pronation -->
<laban:movement>
 <laban:column>5</laban:column>
 <laban:measure>1</laban:measure>
 <laban:beat>0</laban:beat>
 <laban:duration>0.9</laban:duration>
 <laban:hold>false</laban:hold>
 <laban:turn>
  <laban:dynamics>
    <laban:dynamic>strong</laban:dynamic>
  </laban:dynamics>
  <laban:direction>counterClockwise</laban:direction>
  <laban:degree>
   <laban:quantitative>
    <laban:vertical>low</laban:vertical>
    <laban:horizontal>180</laban:horizontal>
   </laban:quantitative>
  </laban:degree>
 </laban:turn>
</laban:movement>

<!-- Supination -->
<laban:movement>
 <laban:column>5</laban:column>
 <laban:measure>1</laban:measure>
 <laban:beat>1.0</laban:beat>
 <laban:duration>0.9</laban:duration>
 <laban:hold>false</laban:hold>
 <laban:turn>
  <laban:dynamics>
   <laban:dynamic>strong</laban:dynamic>
  </laban:dynamics>
  <laban:direction>clockwise</laban:direction>
  <laban:degree>
   <laban:quantitative>
    <laban:vertical>low</laban:vertical>
    <laban:horizontal>180</laban:horizontal>
   </laban:quantitative>
  </laban:degree>
 </laban:turn>
</laban:movement>
```

Listing 7: Movement profile: DoubleFlip interaction technique (excerpt)

The XML code snippet in Listing 7 represents parts of the movement profile. In the listing, both movements of the palm are coded as two movement objects respectively. The profile contains information about the type of movement, timing of the movements, directions, quality of the movements, etc.

Movement profile for Rahman et al. [139] wrist movements interaction technique

Figure 6.9 illustrates the Labanotation representation for utilizing wrist movement for interactions investigated by Rahman et al. [139]. The figure is read as follows: (1) Stand on both feet in a natural stance until the end of the interaction. (2) and (3) The right arm starts at a 90-degree angle to the rest of the body pointing forward and the palm facing up. (4) shows the grasp relationship between the right hand and the interaction object (mobile phone). (5) Complete flexion of the wrist towards the palm side (tilted up at about a 60-degree angle). (6) The wrist returns to the natural neither bent nor flexed position. (7) The last movement is again a complete flexion of the wrist, but this time towards the outer side of the arm, resulting in the hand tilted down at about a 45-degree angle. The corresponding XML representation of the technique is shown in Listing 8. This is an excerpt of the profile. The full representation contains information about the unit of timing (i.e., seconds), the whole duration of the interaction, the default body posture, body parts (i.e., hands), and the interaction's starting position.

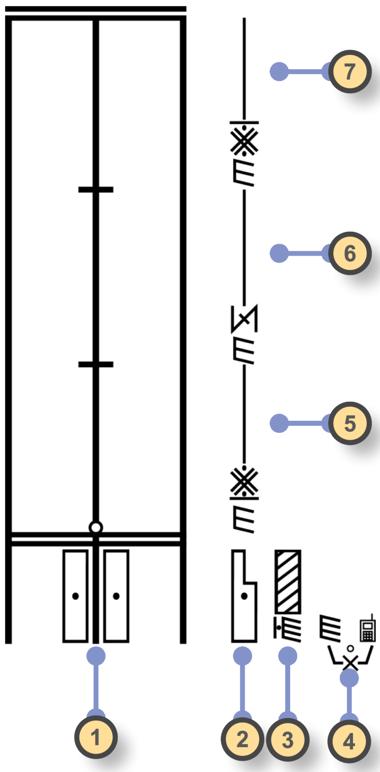


Figure 6.9.: Using Labanotation to document Rahman et al. [139] wrist movements

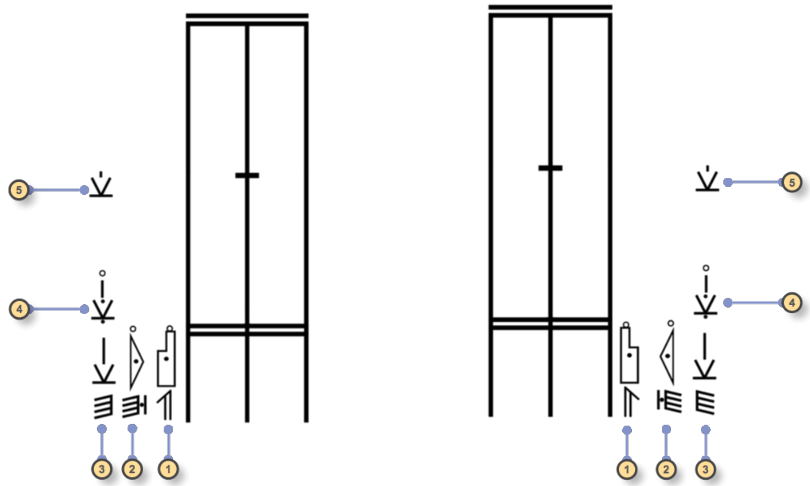
```
<!-- hand tilts up at roughly a 60-degree angle -->
<laban:movement>
 <laban:column>4</laban:column>
 <laban:measure>1</laban:measure>
 <laban:beat>0</laban:beat>
 <laban:duration>1</laban:duration>
 <laban:preSign>
  <laban:bodyPart>
   <laban:joint>
    <laban:joint>wrist</laban:joint>
   </laban:joint>
   <laban:side>right</laban:side>
  </laban:bodyPart>
 </laban:preSign>
 <laban:hold>false</laban:hold>
 <laban:space>
  <laban:spaceMeasurement>
   <laban:type>narrow</laban:type>
   <laban:degree>6</laban:degree>
   <laban:direction>front</laban:direction>
  </laban:spaceMeasurement>
 </laban:space>
</laban:movement>
<!-- hand goes back to normal -->
<laban:movement>
 <laban:column>4</laban:column>
 <laban:measure>1</laban:measure>
 <laban:beat>1</laban:beat>
 <laban:duration>1</laban:duration>
 <laban:hold>false</laban:hold>
 <laban:space>
  <laban:spaceMeasurement>
   <laban:type>none</laban:type>
   <laban:degree>1</laban:degree>
  </laban:spaceMeasurement>
 </laban:space>
</laban:movement>
```

Listing 8: Movement profile: Rahman et al. wrist interaction technique (excerpt)

Movement profile for the 3Gear Pinch interaction techniques

The 3Gear RightPinch and LeftPinch IPs are introduced in section 4.7.1

Figure 6.10(b) and Listing 9 illustrate the modeling of the interaction in Labanotation and interaction profile for the 3Gear right pinch respectively. The Figure 6.10(b) is read as follows: (1) The right arm starts at a 90-degree angle to the rest of the body pointing forward. (2) The palm of the hand points to the left and should remain so during the interaction. (3) The right hand is naturally curved. (4) The right hand is curved and the fingers tips touch each other. The position of the fingers should be held for short time. (5) The hand returns to the natural curve quickly with the fingers naturally spread.



(a) Left hand pinch (b) Right hand pinch

Figure 6.10.: Using Labanotation to Document 3Gear Pinch interaction technique (Left and Right)

Figure 6.10(a) and Listing 10 illustrate the modeling of the interaction in Labanotation and interaction profile for the 3Gear left pinch respectively. The Figure 6.10(a) is read as follows: (1) The right left starts at a 90-degree angle to the rest of the body pointing forward. (2) The palm of the hand points to the right and should remain so during the interaction. (3) The left hand is naturally curved. (4) The left hand is curved and the fingers tips touch each other. The position of the fingers should be held for short time. (5) The hand returns to the natural curve quickly with the fingers naturally spread.

The examples discussed in this section demonstrate the feasibility of documenting and modeling movements for interactions based on movement profiles. Adopting the Labanotation approach for movement profiles requires interaction designers and developers to have good writing and reading skills of the notation. End users are not expected to have the same set of skills. Alternatively, the current machine readable representation of movement profiles can be used to generate useful and more accessible learning material for the end user. For instance, the generation of graphical clues or textual representation of movements becomes possible. Relaying on less standardized forms of movement description (e.g., freeform text description and sketching) reduces the ability for such an automatic documentation generation.

The formal modeling of movements demonstrates the power of accurate dissemination and good record-keeping of interaction techniques. The level of details and abstractions is left out to the designer to decide upon, while keeping the core aspects of the movements involved in the interaction. Once

More discussion about freeform and formalized documentation styles is covered in section 5.5.2

```xml
<laban:movement>
 <laban:column>4</laban:column>
 <laban:measure>0</laban:measure>
 <laban:beat>0.0</laban:beat>
 <laban:duration>1.0</laban:duration>
 <laban:preSign>
 <laban:bodyPart>
   <laban:surface>
    <laban:limb>
     <laban:custom>
     <laban:extremity>
       <laban:joint>
        <laban:joint>hand</laban:joint>
      </laban:joint>
     </laban:extremity>
     </laban:custom>
    </laban:limb>
    <laban:side>inner</laban:side>
   </laban:surface>
   <laban:side>right</laban:side>
 </laban:bodyPart>
 </laban:preSign>
 <laban:hold>false</laban:hold>
 <laban:direction>
  <laban:vertical>middle</laban:vertical>
  <laban:horizontal>left</laban:horizontal>
 </laban:direction>
</laban:movement>
```

Listing 9: Movement profile: 3Gear RightPinch interaction technique (excerpt)

decided, the movements are clearly recorded and represented, leaving very little change for misinterpretations.

6.5. Physical Ability Profile

Physical ability profile is another cornerstone for documenting NI in our conceptual design. It captures the physical abilities required from the user to execute the interaction. It encapsulates information about major life activities vital for the interaction and the importance score for those activities. It also captures the main disabilities that may impact the quality execution of the interactions. This profile invites interaction designers and developers to identify, document, and convey the interaction's essential physical movement needs and properties. The three components can be of great help to ambient spaces to facilitate interaction filtering at runtime to reason about the interactions possibilities at a given context as in [10].

We argue that movement and ability profiles are essential for designing interactions "for all" instead of focusing on a limited population percentile

```
<laban:movement>
 <laban:column>-3</laban:column>
 <laban:measure>0</laban:measure>
 <laban:beat>0.0</laban:beat>
 <laban:duration>1.0</laban:duration>
 <laban:preSign>
 <laban:bodyPart>
  <laban:limb>
    <laban:default>
     <laban:limb>arm</laban:limb>
    </laban:default>
  </laban:limb>
  <laban:side>left</laban:side>
 </laban:bodyPart>
 </laban:preSign>
 <laban:hold>true</laban:hold>
 <laban:direction>
   <laban:vertical>middle</laban:vertical>
   <laban:horizontal>forward</laban:horizontal>
 </laban:direction>
</laban:movement>
```

Listing 10: Movement profile: 3Gear LeftPinch interaction technique (excerpt)

and to avoid inaccessible interfaces. Ability profile (Figure 6.11) contains quantified anthropometric abilities tested by specialists or the user herself. It is defined by three key elements:

- **Physical abilities**: indicate the required physical skills for the interactions, e.g., voluntary movement and range of motion.

- **Physical disabilities** (quantified by impact scores): indicate the quality and duration of a particular disability. Impairment symptoms are normally quantitatively rated with physical assessment and rating scales. Documenting physical disabilities research provides a strong background in this direction. The core matching algorithm in the ensemble engine utilizes then different physical assessment and rating scales to reason about the severity of the symptoms and their impact on the interaction quality.

- **Major life activities**: In our model, each interaction is linked to one or more major life activities such as walking, balancing, seeing, lifting, writing, speech, reaching, dressing, falling, finger tap, arising, etc. The ability to perform the required activity is a good indication on the ability to perform the respective interaction.

Profiling the aforementioned information in abilities profiles is based on the International Classification of Functioning, Disability and Health (ICF)

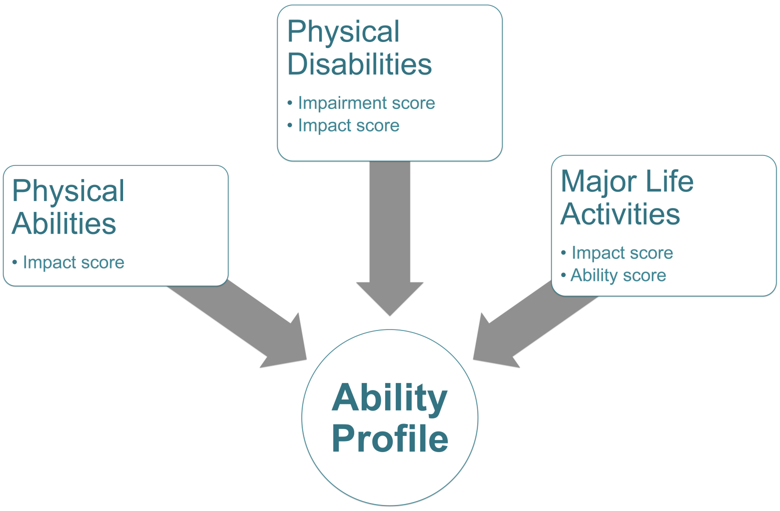


Figure 6.11.: Ability profile components

framework for organizing and documenting information about the functioning and disabilities of the human body [129], proposed by the World Health Organization (WHO). The ICF framework is used to create compatible physical profiles with current practices and standards and enhance the universality of our approach. Moreover, the framework is compatible with the conceptual design of ability profiles introduced in section 3.7.2.

Table 6.1 presents a number of important health related terms defined by ICF.

The ICF framework provides a conceptual ground based on three factors: health condition, personal factors, and environmental factors. ICF structures and classifies the person's health context in terms of: (1) Functioning and Disability and (2) Contextual Factors.

- **Functioning and Disability**: This part deals with two components, namely Body Functions and Body Structures; and Activities and Participation. The evaluation of those components can be positive (expressed by functions, activities, participation) or negative (expressed by impairments, activity limitations, or participation limitations). ICF does not only describe functioning and disability, but also provides ability qualification using qualified measures (used as ability and disability scores in our approach).

- **Contextual Factors**: This part deals with Environmental Factors and Personal Factors. Those factors also follow positive or negative influences (i.e., facilitate or hinder) on the functioning and disabilities. Our ability profile currently considers personal aspects only. Environmental aspects can be applied for environmental physical profiles, a concept that is not covered in this dissertation.

ICF uses an alphanumeric system followed by a numeric code to denote the corresponding components (i.e., b: Body Functions, s: Body Structures, d: Activities and Participation, and e: Environmental Factors) and their detailed subcategories respectively. Additionally, the classification uses a qualifier (a unified generic scale) to denote a magnitude of the level of health. The full specification of various health chapters and subcategories is presented in [129].

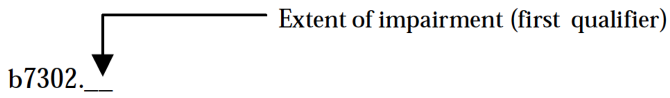
Term	Definition
Functioning	is an umbrella term for body functions, body structures, activities and participation. It denotes the positive aspects of the interaction between an individual (with a health condition) and that individual's contextual factors (environmental and personal factors)
Disability	is an umbrella term for impairments, activity limitations and participation restrictions. It denotes the negative aspects of the interaction between an individual (with a health condition) and that individual's contextual factors (environmental and personal factors).
Body functions	The physiological functions of body systems (including psychological functions).
Body structures	Anatomical parts of the body such as organs, limbs and their components.
Impairments	Problems in body function and structure such as significant deviation or loss.
Activity	The execution of a task or action by an individual.
Participation	Involvement in a life situation.
Activity limitations	Difficulties an individual may have in executing activities.
Participation restrictions	Problems an individual may experience in involvement in life situations.
Environmental factors	The physical, social and attitudinal environment in which people live and conduct their lives. These are either barriers to or facilitators of the person's functioning.

Table 6.1.: ICF health related definitions (according to [129], 212-213)

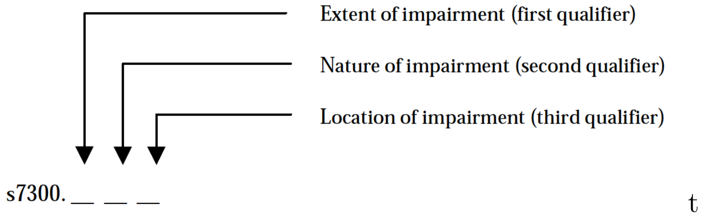
The ICF coding system is illustrated in Figure 6.12. Coding body functions (Figure 6.12(a)) follows simple rules. The first alphanumeric ("b" in this case) corresponds to the component coded ("body functions" in this case). The following numeric code (7302) indicates the corresponding body hierarchy (in this case the "power of muscles of one side of body"). This code is split from the qualifier by a "." mark. The health level is indicated by the qualifier according to a generic scale (0: No impairment, 1: Mild impairment, 2: Moderate impairment, 3: Severe impairment, 4: Complete impairment, 8: not specified, and 9: not applicable.). For instance, the code "b7302.1" means a mild impairment of power of muscles of one side of body and the

Figure E.1, in Appendix D, illustrates the generic qualifier and an example of an ICF-code

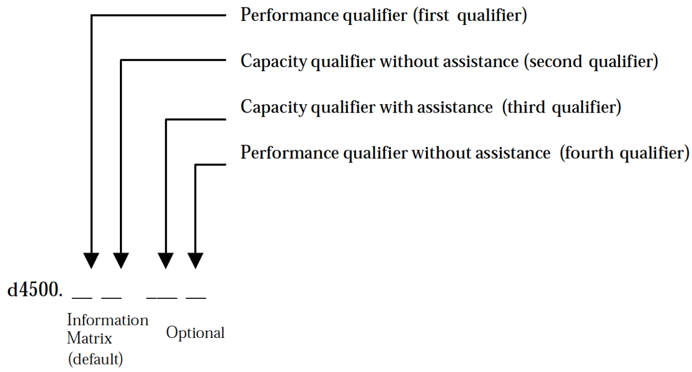
code "b7302.4" means a complete impairment of power of muscles of one side of body.



(a) Coding body funtions



(b) Coding body structures



(c) Coding activities and participations

Figure 6.12.: ICF component-specific coding rules (from [129])

Coding body structure (Figure 6.12(b)) follows the same rules mentioned above. Nonetheless, this component requires three qualifiers. The first qualifier indicates the extent of the impairment (similar to body function coding qualifier). The second qualifier indicates the nature of the impairment (0: no change in structure, 1: total absence, 2: partial absence, 3: additional part, 4: aberrant dimensions, 5: discontinuity, 6: deviating position, 7: qualitative changes in structure, including accumulation of fluid, 8: not specified, 9: not applicable). Moreover, the third qualifier indicates the location of the impairment (0: more than one region, 1: right, 2: left, 3: both sides, 4: front, 5: back, 6: proximal, 7: distal, 8: not specified, 9: not applicable).

Similarly, coding activities and participation (Figure 6.12(c)) follows the same rules with some variations in the qualifier section. Activities and participation component requires two qualifiers (Performance and Capacity without assistance, as the first and second qualifiers respectively). Additionally, two other optional qualifiers can be used to indicate the capacity and performance with assistance respectively. The qualifiers follow the same scale (0: No difficulty, 1: Mild difficulty, 2: Moderate difficulty, 3: Severe difficulty, 4: Complete difficulty, 8: not specified, 9: not applicable). For instance, "d4500" indicates walking short distances. "d4500.2 _" indicates moderate restriction in performance of walking short distances.

A full description of the ICF coding is out of the scope of this dissertation, please refer to the ICF specification document [129] for detailed description and more examples.

A general overview over the ability profile XML scheme is presented in Figure 6.13. A detailed overview over the ability element is presented in Figure 6.14.

Appendix E presents a full overview over the ability profile XML scheme

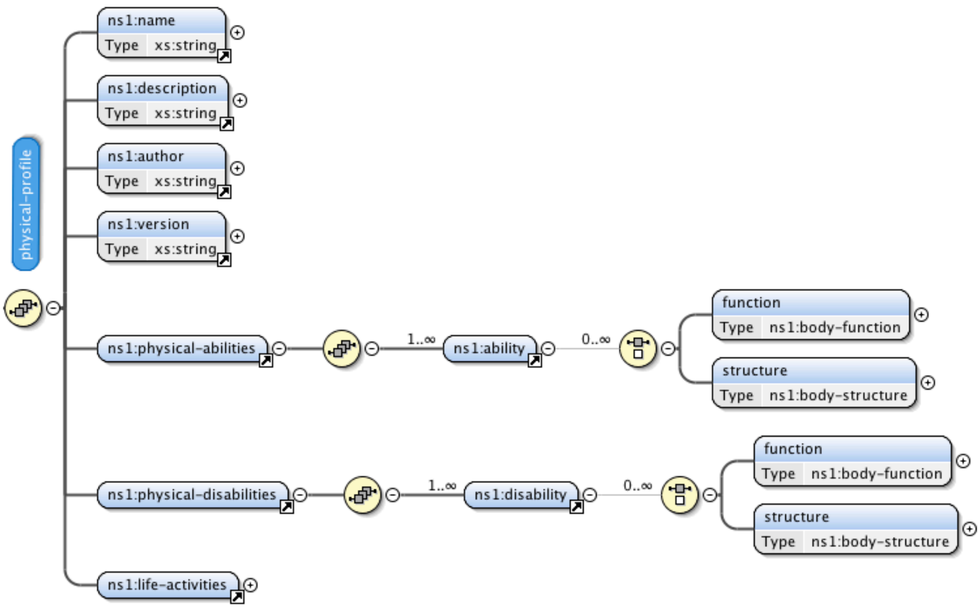


Figure 6.13.: Ability profile scheme (high-level overview). The full overview over the ability profile is presented in Appendix E

We believe that ability profiles strongly benefit from the ICF as a widely adopted interactional classification [129]. The framework provides a standardized and common language for communicating health related information across domains, especially with its systematic coding scheme. This allows the profiles to be compatible with various medical and none-medical services and domains (e.g., health-related diagnosis services and health providers).

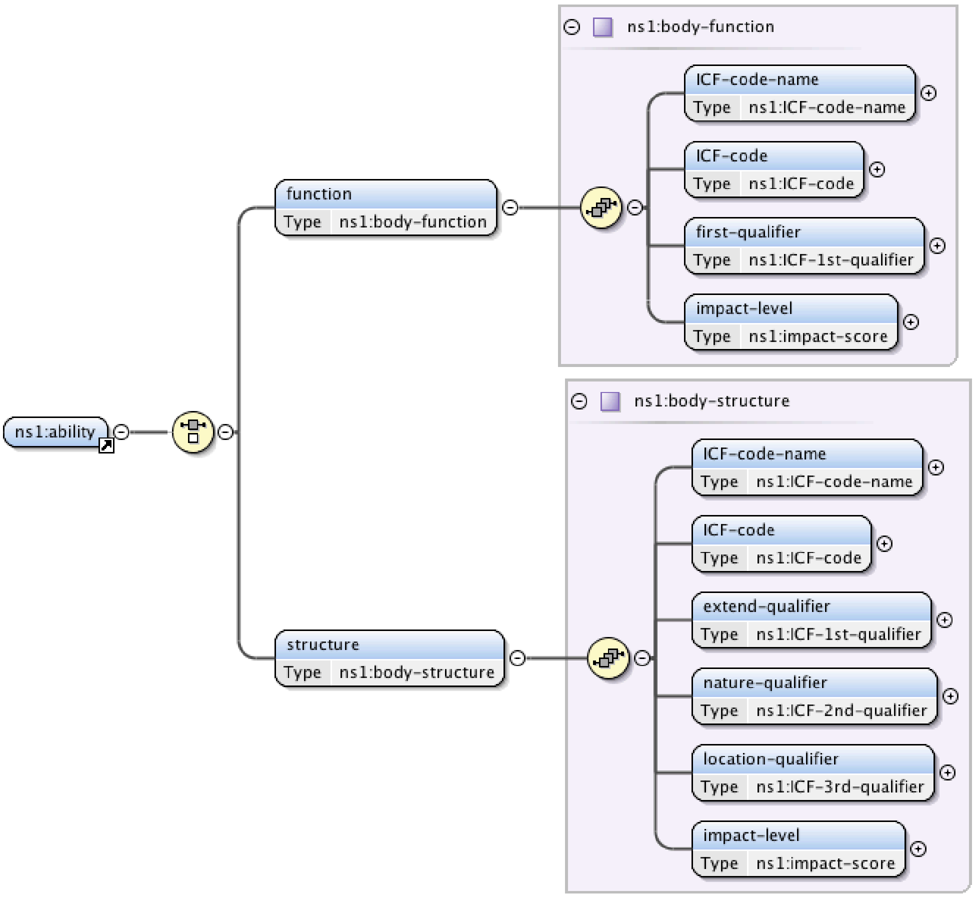


Figure 6.14.: Ability profile scheme - physical ability element (high-level overview). The full overview over the ability profile is presented in Appendix E

Our current implementation of ability profiles does not capture the full scope of the ICF framework, instead our current scheme covers merely the functioning and disability components. Originally, the ICF covers also a second part, namely the contextual factors (including environmental and personal factors). The contextual factors are concerned about any personal or environmental factor that may have an impact on the person's health conditions or states. For instance, the home and workplace settings are considered as part of the individual environmental factors.

Our conceptual design of ability profiles does not require the source or the cause of the user's physical abilities or disabilities. Although such information may be relevant for ambient systems generally, we have opted not to include such information in the ability profile scheme, mainly because it does not relate directly to the scope of interaction techniques and it will increase the complexity of physical profiling.

6.5.1. Example and Showcases

In this section, the ability profile of the 3Gear Right Hand Pinch interaction technique is presented and briefly discussed.

Ability profile for the 3Gear Right Hand Pinch interaction technique

The ability profile shown in Listing 11 defines three mainly physical qualities based on ICF. First, the profile requires one ability, namely a coordination of voluntary movements (ICF code - b7602). Indicated by the first qualifier, the technique tolerances mild impairments (indicated by the value "1"). The importance of this physical ability is indicated by the impact score, with the value "1" indicating a very a high impact on the interaction.

Second, the profile includes information about one structural disability that impact the interaction, namely structure of joints of hand and fingers (ICF code - s73021). The value "4" in the extend qualifier indicates a sever impairment, which is caused by the absence of the structure (indicated by the value "1" in the nature qualifier). Moreover, the location qualifier indicates the impairment in the the right side of the body (i.e., right hand and fingers). This disability impact the interaction severely as indicated with a high impact score.

Third, the profile defines hand and arm use (ICF code - d445) as a required life activity for the interaction. Nonetheless, the interaction tolerates mild difficulties (value "1" in the performance-qualifier) and with smaller impact score than the two previous qualities.

6.6. Interaction Editor

The section 6.2 has covered an analysis about existing documentation related tools for documenting and authoring movements for interactions in NUIs. The current absence of tools and standard methods turns documenting interactions into a very tedious process. Therefore, we have integrated our approach into an authoring tool, called the Interaction Editor for documenting and describing movements for interactions.

The described part in this section was completed in close collaboration with Jan Gröschner, a Bachelor's degree candidate, during his thesis [63]

The editor is implemented to offer interaction designers the ability to build Movement Profiles according to the design presented in section 6.4. The

```xml
<?xml version="1.0" encoding="UTF-8"?>
<physical-profile xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 xsi:schemaLocation="http://www.w3schools.com abilityprofile.xsd"
 xmlns="http://www.w3schools.com">
 <name>3Gear Right pinch</name>
 <description>he 3Gear right pinch interaction requires the user
  perform a pinching gesture with the right arm. </description>
 <author>Bashar Altakrouri</author>
 <version>1.0</version>
 <physical-abilities>
  <ability>
   <function>
    <ICF-code-name>Coordination of voluntary movements</ICF-code-name>
    <ICF-code>b7602</ICF-code>
    <first-qualifier>1</first-qualifier>
    <impact-level>1</impact-level>
   </function>
  </ability>
 </physical-abilities>
 <physical-disabilities>
  <structure>
   <ICF-code-name>Joints of hand and fingers</ICF-code-name>
   <ICF-code>s73021</ICF-code>
   <extend-qualifier>4</extend-qualifier>
   <nature-qualifier>1</nature-qualifier>
   <location-qualifier>1</location-qualifier>
   <impact-level>1</impact-level>
  </structure>
 </disability>
 </physical-disabilities>
  <life-activities>
   <activity>
    <ICF-code-name>Hand and arm use</ICF-code-name>
    <ICF-code>d445</ICF-code>
    <performance-qualifier>1</performance-qualifier>
    <capacity-without-qualifier></capacity-without-qualifier>
    <performance-without-qualifier></performance-without-qualifier>
    <capacity-with-qualifier></capacity-with-qualifier>
    <impact-level>0.7</impact-level>
   </activity>
  </life-activities>
</physical-profile>
```

Listing 11: Ability profile: 3Gear RightPinch interaction technique

editor enables a bi-directional conversion between the machine readable XML representation and the human readable Labanotation visual notation. With its graphical capabilities, designers can use the editor to design their movement profiles visually. For examples, designers can construct new movements by dragging and dropping the visual representation of the movement part and easily editing existing movements components (e.g., drag the edge of a movement object to change timing). Moreover, designers may construct movement profiles by coding the profile in XML format using the XML view. The changes will be then reflected visually when activating the visual view.

The following sections will cover an overview over the editor's requirements, introduce its architecture, and discuss its implementation.

6.6.1. Requirements and design choices

This section presents the functional requirements identified for implementing the editor, the absence of any of those features impacts the utility of the editor:

- **Visualizing the Labanotation staff**: The editor is required to visualize non-standard column definitions, measures, and beats. The staff visualization aids the design to correctly place movement components and timing information in the profile.

- **Visualizing major related Labanotation symbols**: Adequate graphical representations of the major Labanotation symbols are required. According the current design of the movements profile covered in section 6.4.2, only a subset of Labanotation symbols are supported including: path symbols, room direction signs, direction symbols, turn symbols, a cancel sign, a return-to-normal sign, a vibration symbol, space measurement signs, and signs for major body and customized joints.

- **Direct manipulation of the movement profile**: The editor allows the editor's user to directly edit, compose, and manipulate the movement profile visually. The user should be able to select required movement components from a palette and simply drop the components into adequate destinations. Moreover, the user should be able to select any component in the score and apply manipulation actions directly (e.g., scaling and repositioning components).

- **Textual manipulation of the movement profile**: It should be possible to edit and view the profile components in textual form by manually manipulating and changing their properties, especially the timing and placement information.

- **Bi-directional conversion between visual Labanotation representation and the movement profile XML scheme**: Any changes made on either representation should be reflected on the other. Both XML and visual representations should be available to be viewed and permanently saved using the editor.

- **Simplified wizard-based creation of movement profiles:** The editor should provide simple and easy to use templates for creating movement profiles.

- **Multi-platform support**: The editor should support major desktop-platforms including: Windows, Linux and Mac OS X.

- **Loading movement profiles from external resources:** Any XML file complying with the movement profile scheme can be loaded to the editor, even if it was created by an external editor.

- **Support for undo and redo actions:** The editor should provide the undo and redo functions, commonly found on most editors.

6.6.2. Architecture

The Interaction Editor was build based on Eclipse plugin architecture that guarantees an extensibility and cross platform functionality. The Eclipse platform provides a very strong modular approach for designing, extending, and integrating plugins. Moreover, it provides useful and simple APIs for handling wide range of actions and operations such as storage management, properties views, and undo/redo-operations. Principally, the editor is build based on the Eclipse Graphical Editing Framework (GEF), which provides the bases for building graphical editors and strongly relies on Model-View-Controller architecture.

Figure 6.15 illustrates the general overview over the editor's block diagram. The design of the editor follows the Model-View-Controller pattern recommended by GEF.

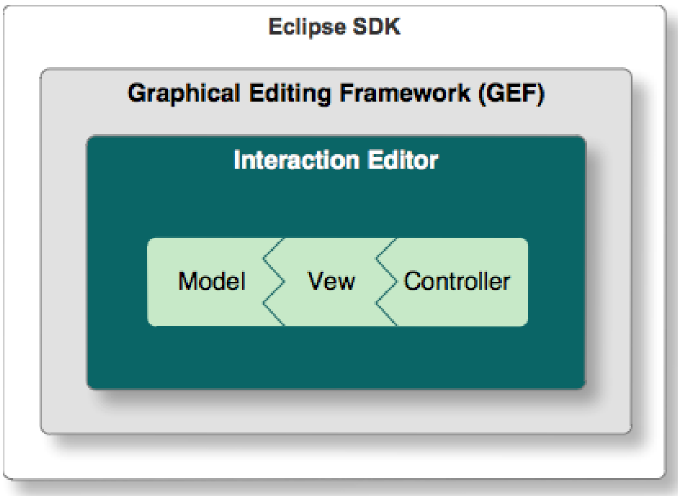


Figure 6.15.: Interaction Editor high-level architectural block diagram overview

Model

The Model-View-Controller pattern requires all graphical components of the editor to have an internal model representation. We have designed the representation for the movements profile elements based on their corresponding XML representation we have defined in section 6.4.2. Each of the Labanotation graphical elements are represented by an own class. Additionally, we have provided each class with a dedicated serialization method to serialize and deserialize the object from and its XML representation. This is an essential functionality required by GEF. For instance, Listing 12 shows an excerpt from the Staff-class serialization method and listing 13 shows an excerpt from its deserialization method.

```
public String toXmlString(String prefix) {
 String xml = prefix + "<laban:measures>" + measures
  + "</laban:measures>\n" + prefix + "<laban:timing>\n"
  + timing.toXmlString(prefix + "   ") + "\n" + prefix
  + "</laban:timing>\n";
  if (columnDefinitions != null && columnDefinitions.size() > 0) {
   xml = xml + prefix + "<laban:columns>\n";
   for (ColumnDefinition colDef : columnDefinitions) {
    xml = xml + prefix + "   " + "<laban:columnDefinition>\n"
    + colDef.toXmlString(prefix + "      ") + "\n" + prefix
    + "   " + "</laban:columnDefinition>\n";
    }
   xml += prefix + "</laban:columns>\n";
   }
  xml += prefix + "<laban:movements>\n";
  if (movements != null && movements.size() > 0) {
   for (IMovement mov : movements) {
    xml += mov.toXmlString(prefix + "   ") + "\n";
   }
  }
  xml += prefix + "</laban:movements>";
  return xml;
}
```

Listing 12: The Staff-class' toXmlString-method (excerpt)

GEF requires each of the internal models to extends the "java.util.Observable" class and implement its "java.util.Observer" interface. This will provide a notification functionality for the corresponding controller classes. The "MessageConstants" interface, used by the "Observable.notifyObservers" method, allows the controller to distinguish between events.

The main classes that extend the Observable-class are the Score-class (which is the root class of the model to apply changes to a score's meta-data back to the interface such as authors' names, the score's title, and the score's description), ColumnDefinition-class (to apply changes to a column's

```java
public static Staff buildFromXML(XMLStreamReader parser) {
 int measures = 0;
 Timing timing = null;
 List<ColumnDefinition> colDefs = null;
 List<IMovement> movements = null;
 try {
  for (int event = parser.next(); !(event == XMLStreamConstants.END_ELEMENT && parser
  .getLocalName() == "staff"); event = parser.next()) {
   switch (event) {
    case XMLStreamConstants.START_ELEMENT:
    if (parser.getLocalName() == "measures") {
     measures = Integer.parseInt(parser.getElementText());
    } else if (parser.getLocalName() == "timing") {
     timing = Timing.buildFromXML(parser);
    } else if (parser.getLocalName() == "columns") {
     colDefs = ColumnDefinition.buildFromXML(parser);
    } else if (parser.getLocalName() == "movements") {
     movements = parseMovementsElement(parser);
    }
   break;
   }
  }
 } catch (XMLStreamException e) {
  e.printStackTrace();
  }
 if (colDefs == null) {
  colDefs = new LinkedList<ColumnDefinition>();
 }
 if (movements == null) {
  movements = new LinkedList<IMovement>();
 }
 Staff result = new Staff(measures, timing, colDefs, movements);
 result.configureMovements();
 return result;
}
```

Listing 13: The Staff-class' buildFromXml-method (excerpt)

pre-sign), Staff-class (to apply changes on the actual movement represented in a Labanotation score), Movement-class (to apply changes to the position and the duration of movement including flexion and extension), Path-class, Relationship-class (to apply changes to the user's relationship to objects or persons), and RoomDirection-class (to apply changes related to the physical surrounds).

View

The graphical representation for all GUI interface components is implemented based on Draw2d, which is a lightweight graphics toolkit used heavily by GEF. Each of the Labanotation symbols in the editor is composed from nested Draw2d components (consisting from a range of simple primitive shapes such as rectangles, ellipses, and polygons to more complex figures created from bitmap-files). Due to the lack of vector support in GEF, all resizable symbols in the editor were recreated in source-code in order to guarantee the rendering quality of the symbol or element when scaled up or down. The editor utilizes a custom made StaffLayoutmanager-class to match the model component's hierarchy on the view component. This class is responsible for snapping and arranging graphical symbols of the Labanotation symbols in the staff correctly. Such functionality is not available in the conventional Draw2d LayoutManagers class, thus had to be implemented.

Controller

The editor's controller component consist of the main following class types:

- **EditParts**: This class type is responsible to track any model changes and to update the visual representation accordingly. It is also responsible to track any user-interaction back into the model. In our implementation, each of the model components (i.e., classes) has a corresponding "EditParts" class. Importantly, each of those classes implements that "java.util.Observer" interface to reflect any changes to the model or interface. Internally, the GEF creates a runtime instance for each of the "EditPart" objects. Once created, the object is registered as an observer with its model-instance and a graphical presentation appears on the editor view. Any changes on the view (due user interactions with the editor) is mediated to the "EditPart" instance through a dedicated "update" method.

- **EditPolicies**: This type of classes is responsible to changing the model presentation based on the user interaction with the graphical interface. While GEF contains various types of "EditPolicies", the editor is currently using "ComponentEditPolicies", "ResizableEditPolicies", and "LayoutEditPolicies. The "ComponentEditPolicy" allows removing a model-instance. The "ResizableEditPolicies" takes care of resizing Labanotation symbols in the GUI view. The "StaffLayoutEditPolicy" handles the changes in the model. In the case of resizing, it handles an "EditPolicy" and creates a command instance to change the duration of Labanotation movement.

- **Commands**: This type of classes is usually responsible to change a targeted model-instance or its attributes. Commands are written in Eclipse command stack, which saves all the information necessary to redo and undo operations.

6.6.3. Interacting with the Interaction Editor

The editor can be used on most available platforms and it has been tested on Windows 7, Fedora 16 and Mac OS X 10.6 using Eclipse 3.7. In this section we will illustrate the use and interaction of the the editor using a sequence of screenshots. The modeled example is the 3Gear right hand pinch (presented in section 6.4.3).

Shown in Figure 6.16, the editor provides a simple wizard for easy and practical creation of movement profiles. The creation of any movement profile follows two steps. First, the interaction designer creates a new Eclipse file of type "Movement Profile" from the Interaction Editor menu. Next, the designer creates the Labanotation score for the profile including the title of the interaction, the description, the author(s), score timing information, and the storage location of the profile.

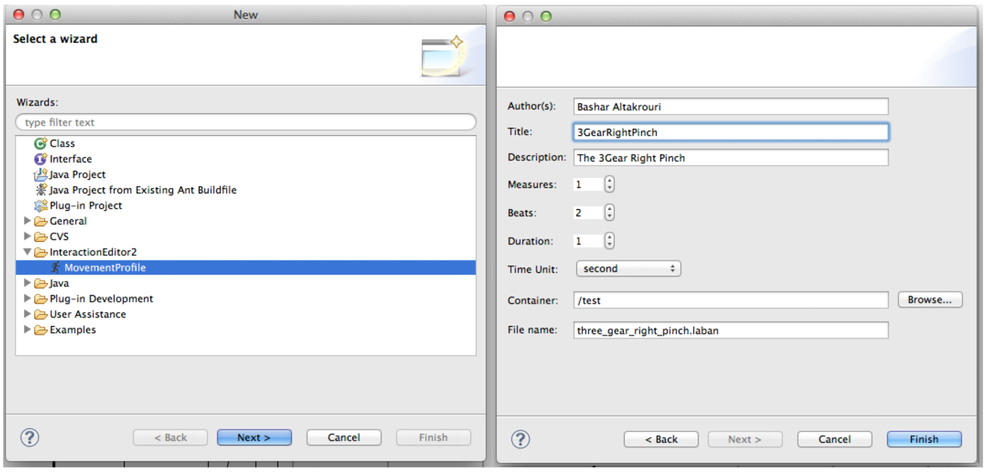


Figure 6.16.: Interaction Editor: Eclipse New file wizard (left) and new Labanotation score wizard (right)

Once completed, the editor's main view is presented to the designer as shown in Figure 6.17.

The figure shows the main GUI views and elements of the editor. The editor is split into the following three main views:

- Main View: This view is the main drawing area of the movement profile (i.e., mainly reserved for the Labanotation score).

- Labanotation Palette: This view provides all the Labanotation symbols used to draw the score such as body parts, direction symbols, and relationship symbols. The editor provides representations for the major limb pre-signs (i.e., arm, leg, neck, upper arm, lower arm, thigh and lower

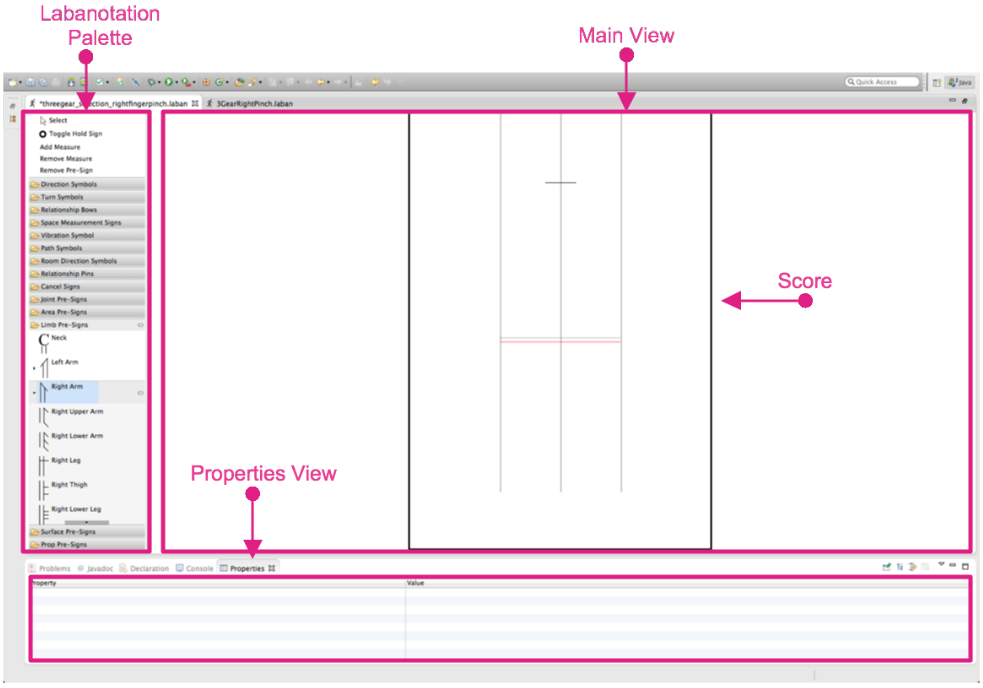


Figure 6.17.: Interaction Editor: A documentation tool for authoring movement profiles

leg). This features allows for short movement inclusion of those limbs. The designer can select any item and drop it into its adequate position in the score. The palette contains space measurement signs. Beyond expressing flexion, extension, and bending, those signs can be used to modify other symbols (e.g., direction symbols). In many cases modifying symbols reduces the complexity of specifying positions for individual joints.

- Properties View: This view provides the detailed properties (especially symbol's measure, beat, column and duration) for any selected item in the score. For instance, the designer may use this view to change the scale or the timing for any symbol in the score.

To build the 3Gear right pinch movement profile, the designer selects the appropriate Labanotation symbols from the Labanotation Palette into the score. Figure 6.18 illustrates the designer's action to add a column for the right arm by dropping the arm symbol into the right side of the score. Similarly, all movement components of the profile are added in the same way. In Figure 6.19, we illustrate dropping the folding symbol to return the hand into the natural curve quickly.

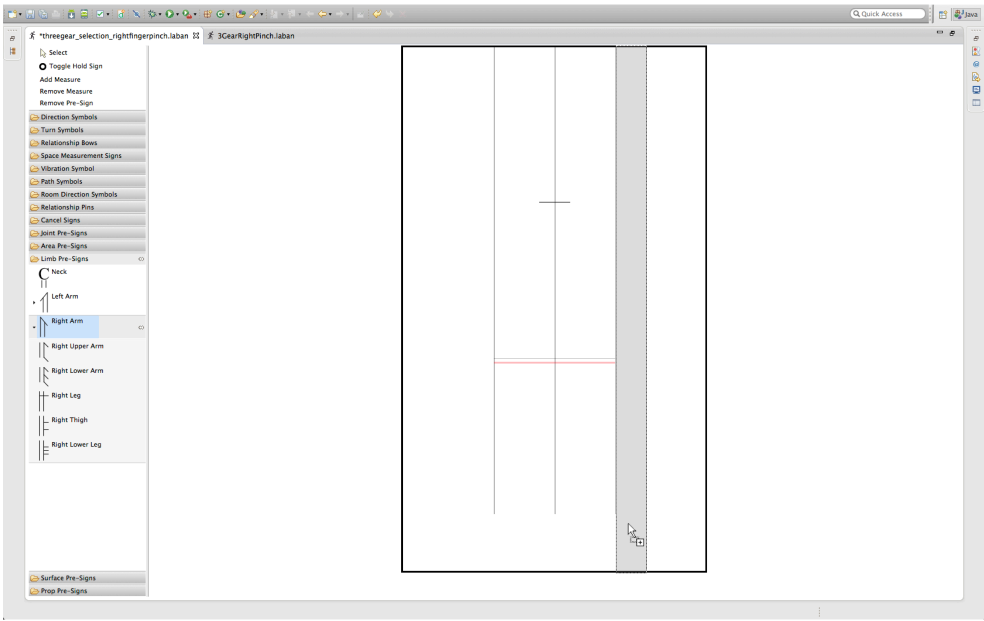


Figure 6.18.: Interaction Editor: Adding a column definitions to the score (This figure appears magnified in Figure G.1 in Appendix G)

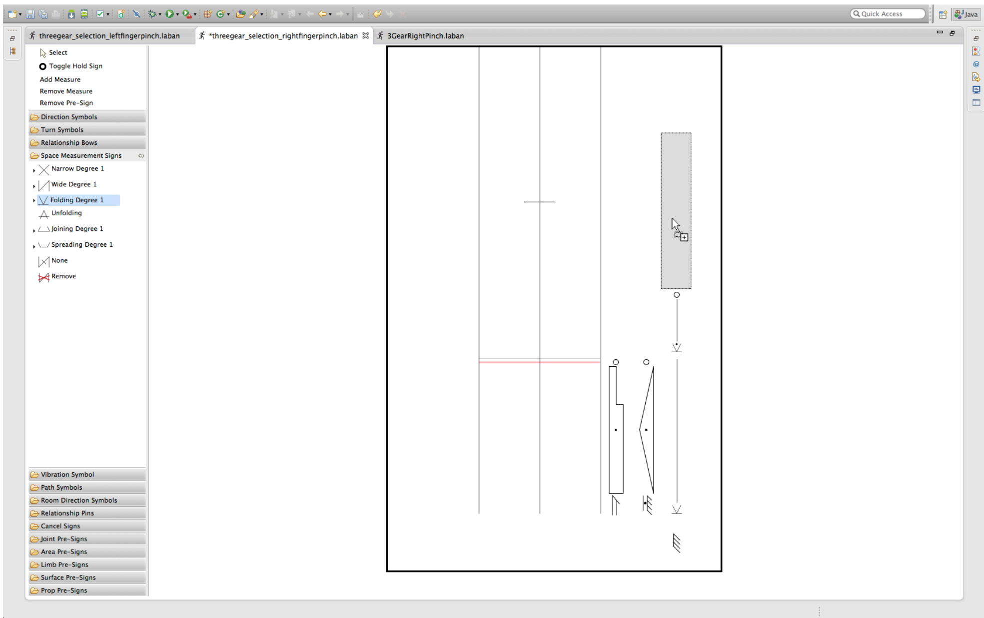


Figure 6.19.: Interaction Editor: Adding a folding symbol to the score (This figure appears magnified in Figure G.2 in Appendix G)

Symbols can be edited by direct visual manipulation of the symbols on the screen (e.g., dragging the symbols side up or down for scaling) or by editing its properties from the properties view (shown in Figure 6.20).

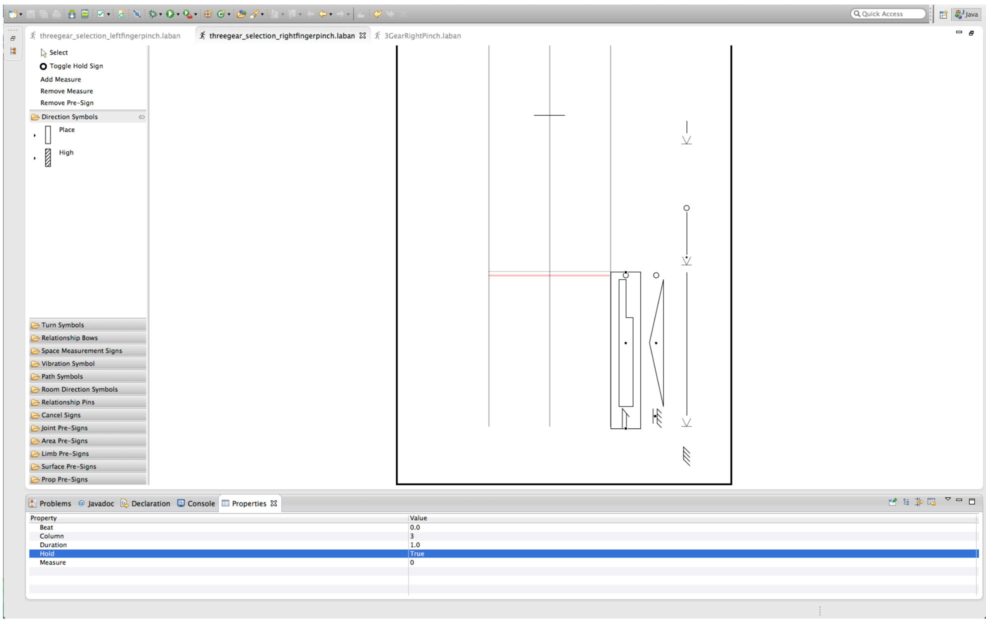


Figure 6.20.: Interaction Editor: Changing the properties of a movement component (This figure appears magnified in Figure G.3 in Appendix G)

Using symbols' timing information (i.e., measure, beat and duration, and column index), the editor places and resizes the major symbols on the staff according to the provided information. Direct visual manipulation of any graphical symbol in the staff (e.g., changes the size and position of a symbol) is reflected on the internal symbol attributes. To remove a particular symbol, the user should select a component using the mouse or the keyboard and invoking the delete command. This is achievable by hitting the delete-key, selecting the delete-option of the context-menu, or using the delete-button provided on the menu-bar (shown in Figure 6.21).

Figure 6.22 shows the complete Labanotation score for the movement profile with all symbols snapped into their adequate locations on the score. Figure 6.23 shows the XML representation of the modeled profile.

6.7. Conclusion

In this chapter, we have argued that interaction documentation is an important aspect of successful adoption of NUI in ambient spaces. NUI

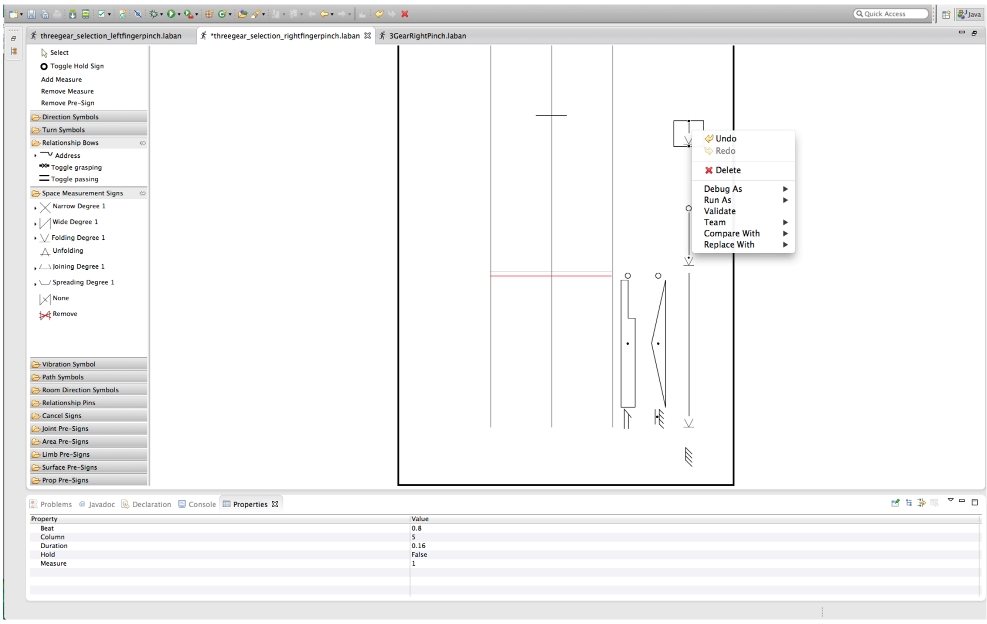


Figure 6.21.: Interaction Editor: Deleting a movement component form the score (This figure appears magnified in Figure G.4 in Appendix G)

documentation as a decimation and sharing strategy for NUI is a primary cornerstone for the concept of Interaction Ensembles. Moreover, this dissertation argues that documentation is not only essential for sharing and deploying interactions, but also to aid the process of learning new interactions and recalling known NI interactions. Norman [126] stressed the fact that many developed interaction techniques are novel and unknown to users, hence learning Interaction Ensembles is inherently a much more challenging task.

Currently, interaction documentation is not fully suitable for ambient spaces and does not adequately satisfy their challenging qualities. Moreover, the chapter reviews the existing documentation-related tools and reveals four general observations including the lack of widely adopted tools by NUI designers, the absences of dedicated NUI documentation tools, the lack of end-user support, and the lack of support and considerations of body movements and postures as part of the interaction descriptions (if at all found). In fact, most of the discussed studies cited in this section do neither target NI documentation per se nor provide a general methodology. Hence, the chapter strongly indicates that the issues of NUI documentation remain a challenging and unresting task for the HCI community.

Although documentation may be achieved in different forms and considered different factors, we have based our approach on centralizing interactions around the human body and its movements. The chapter presents a review

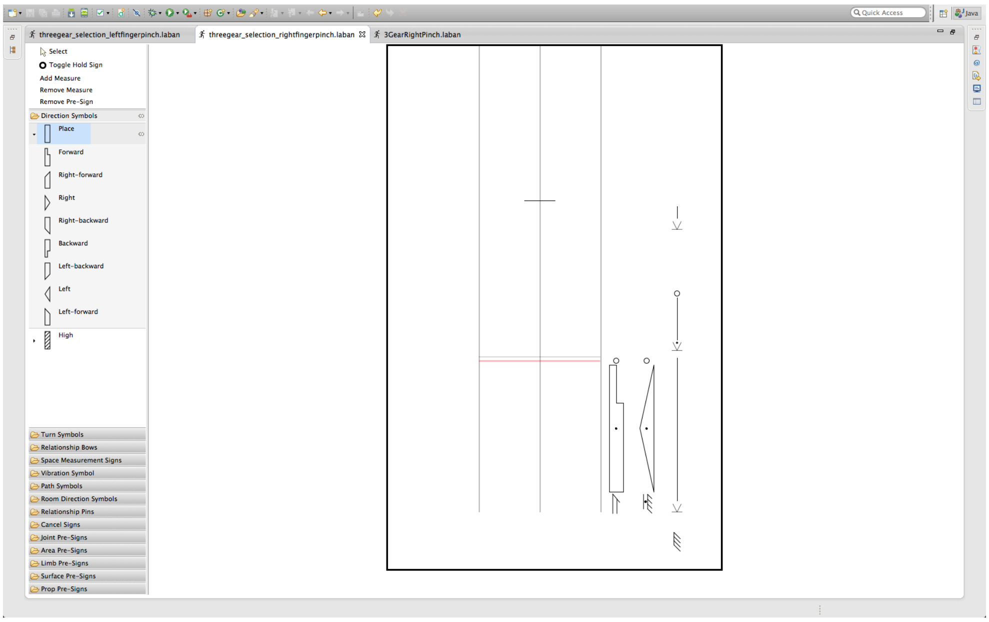


Figure 6.22.: Interaction Editor: Complete Movement Profile in Labanotation visual representation (This figure appears magnified in Figure G.5 in Appendix G)

on movement documentation as an unresting problem for many fields including HCI. Moreover, the chapter discusses the extra challenges that ambient spaces impose on describing interactions including the heterogeneity of users' needs and abilities, heterogeneity of environment context, and media renderers availability.

The suggested documentation approach in this chapter is based on documenting movement qualities and physical ability qualities, by modeling those qualities in movement profiles and physical profiles respectively. The chapter discusses those two profiles, provides various modeled examples, and illustrates their technical implementation in the STAGE vision and currently working system. Moreover, the chapter presents an interaction documentation tool called the Interaction Editor as part of our vision and effort to fulfill and solve this research gap. The editor can be used by interaction designers to create movement profiles. The use of the editor simplifies the IP profiling process and reduces the time required for implementation.

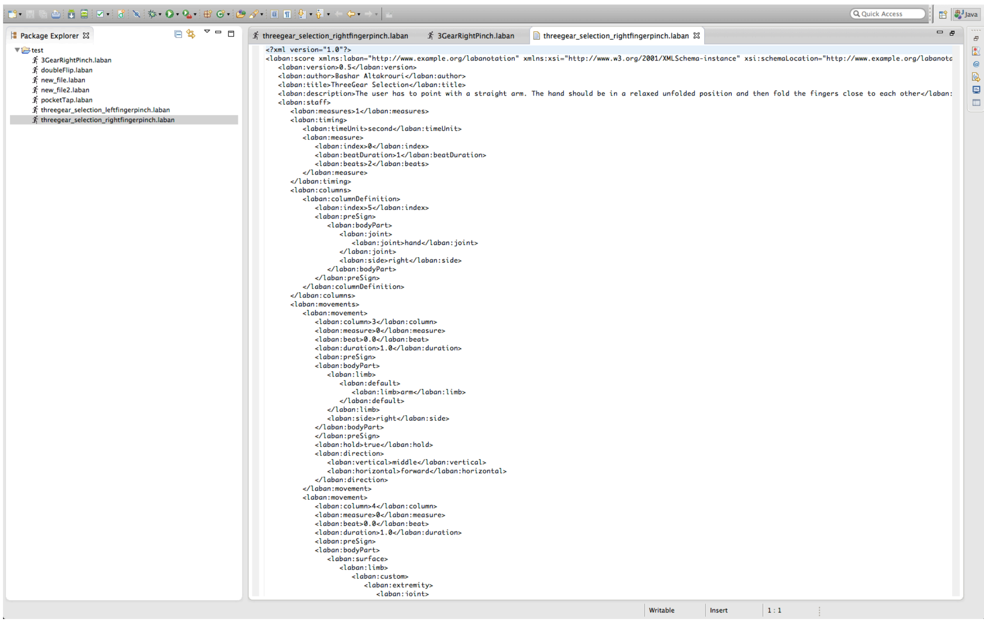


Figure 6.23.: Interaction Editor: Complete Movement Profile in XML representation (This figure appears magnified in Figure G.6 in Appendix G)

7 Conclusions

This chapter presents the key conclusions, results, and contributions achieved by the work described in this thesis. This thesis contributes to the HCI community effort towards utilizing the whole human body in motion for better integrated interactions in ambient systems. The motivation behind this thesis is threefold. Firstly, it is motivated by a plethora of novel Natural Interaction techniques proposed each year for an engaging, enhanced, and effective interaction experience, but many of them are lacking a holistic support for a Whole Body interaction engagement. Secondly, it is also motivated by increasing diversity of user population in interactive systems, caused by the demographic change towards an older population, inclusion of child support in interactions, and more support for physically challenged users. This diversity widens the heterogeneity of the population's physical abilities and challenges the design, implementation, and deployment of interaction techniques and interactive systems. Thirdly, the thesis is motivated by the increasing necessity for in-situ deployment and adaptation of interaction techniques in ambient spaces, based on the idea of de-coupling the close binding between devices, interaction techniques, and applications.

While the bulk of HCI research in this field is targeted at designing novel interaction techniques, evaluating the usability of interactions, and investigating micro-scale interaction adaptation (focusing on single isolated interactions), this thesis is one of few research projects targeting macro-scale interaction deployment and adaptation in ambient spaces. This thesis strives to propose a new conceptual framework for NUI deployment, adaptation, and sharing in ambient spaces. In addition to investigating the challenges and issues of this vision, a reference model, architecture, and implementation are proposed.

This thesis proposes *Interaction Ensembles* as a novel conceptual model for NUI adoption and integration in ambient spaces. *Interaction Ensembles* foster the idea of tailoring multiple interaction modalities at runtime to maximize the adoption of available interaction resources and possibilities to the user's physical abilities, needs, and context. By using Interaction Plugins, the approach opens many important questions regarding developing, deploying, adopting, and sharing interaction techniques. *Interaction Ensembles* may provide novel and powerful means of interactions in ambient spaces. Similarly, ensembles can be applied to closely related fields of research such as tangible interactions.

Investigating this concept has revealed various research gaps and a large number of open research questions, out of which the following three research issues are covered extensively in this dissertation:

- **Issue 1.** Deploying and sharing Natural Interaction techniques based on the users' physical abilities and context in ambient spaces.

- **Issue 2.** Authoring body movement, physical abilities, and physical disabilities as the central anchors of NUIs in ambient spaces.

- **Issue 3.** Investigating the current NUI documentation and learning practices, as well as their impact on the proposed conceptual framework.

While many other research issues are of relevance and importance to foundations of the thesis vision, we have opted for the three aforementioned issues as a starting point in this novel area of research. Section 7.1 presents the main achievements and contributions of this thesis towards the three aforementioned research issues. Section 7.2 presents the main open issues and various future research directions related to the thesis's vision.

7.1. Achievements and contributions

The thesis features a number of achievements and contributions. Herein, we summarize the most relevant and important contributions.

7.1.1. NI ensembles

The thesis proposes NI Ensembles as an anthropometric framework for NUI adoption and adaption in ambient spaces. This approach is designed and presented as a conceptual model to overcome the limited support for full body exploration for NUI. The proposed conceptual design fosters soft-wiring applications and interaction devices, whereby minimizing the limitations of the conventional static binding of applications and devices. Hence, the approach aims at eliminating the mismatch between the users' needs and the interaction offerings in ambient spaces. The conceptual design is based on Interaction Plugin design, anthropometric driving matching and profiling of NUI, on-demand wiring of interaction resources, and community-based designing and sharing of NUI.

The work on this conceptual model was mainly targeted at defining the *Interaction Ensembles* concept, standalone deployable interaction objects (Interaction Plugins), and six different useful cases for *Interaction Ensemble*. Additionally, the model is targeted at understanding the integration of those concepts in the design of ambient spaces.

7.1.2. Deploying and sharing Interaction Plugins in ambient spaces

Self-contained interaction objects, called Interaction Plugins, are proposed to enable interaction sharing and runtime deployment, based on three main design characteristics: matching users and NI physical context, precise and extensible NI description, which is made human and machine readable, and flexible deployment of NI techniques at runtime.

The STAGE technical implementation is presented for interaction and ensemble deployment in ambient spaces. STAGE presents one of many possible reference implementations and architectures to realize the conceptual design proposed by the thesis. In STAGE, IPs are implemented to enable interaction sharing and runtime deployment on Android powered mobile devices. The implementation allows interaction techniques to be deployed at runtime based on the users' physical abilities and interaction needs of the interactive applications.

To demonstrate some aspects of the described runtime deployment of IPs, the AmbientRoom interactive application is implemented as a living demonstration of how mobile devices may become interaction hubs to control the room's lighting based on hand gestures. Currently, the demonstrator illustrates the implementation of IPs, IP dynamic deployment, IP filtering based on interaction, ability, and movements profiles, and an interactive demo application that utilizes a few IPs dynamically at runtime.

7.1.3. Profiling NI techniques' physical qualities

We have conceptually designed movement profiles, interaction profiles, and ability profiles as cornerstones for an anthropometric driven matching and presentation of IPs.

Following the main approach of this thesis by centralizing interactions around the human body and its movements, the thesis argues that body movements and physical abilities are central anchors of context information for Interaction Plugins in ambient spaces. Such context information is of high importance in ambient spaces because of the heterogeneity of users' needs and abilities, heterogeneity of environment context, and media renderers' availability.

In STAGE, important context qualities are modeled in terms of movement and physical profiles covering movement qualities and physical ability qualities respectively. Firstly, movement profiles capture the movement components and qualities of the interaction, for instance the body part, type of movement, timing, etc., involved in the interaction. This profile is based on Labanotation as one of the most flexible and yet comprehensive physical movement recording and analyzing systems. A compliant XML-based model is designed and implemented to satisfy machine and human readability. Moreover, an interaction authoring tool, called Interaction Editor, is suggested to the design and development processes of movement profiles visually or in XML format. The movement profile is used in STAGE during the interaction runtime deployment to reason and filter interaction based on the users' abilities and disabilities, as well as the required interaction semantics.

Secondly, the ability profile is designed and implemented to capture the physical abilities of the user. The profile features three important context information, namely Physical Qualities to indicate the required physical skills for the interactions, e.g., voluntary movement and range of motion; Physical Disabilities to indicate the quality and duration of a particular disability; and Life Activities, e.g., walking, balancing, seeing, lifting, to indicate the ability to perform the required activity as a good indication on the ability to perform the respective interaction. Profiling abilities is based on the ICF framework proposed by the WHO in order to increase the compatibility of the physical profiles with current most used approach.

7.1.4. NUI documentation and learning practices

We have presented a novel fourfold investigation on NUI documentation and learning practices. The investigation included (1) an online exploratory survey on documenting Natural User Interfaces (NUI) answered by 64 designers and 267 end users; (2) coding and analyzing a sample of 93 recently ACM published multitouch and motion-based interaction papers; (3) coding and analyzing three new motion-based applications market initiatives; and, (4) reviewing a number of NUI documentation-related tools and languages.

The investigation included a number of observations. Firstly, it presented the NUI designers' most commonly applied documentation choices, most importantly, documentation frequency and media type of choice. Secondly, it presented the NUI users' learning habits and preferences, most importantly, learning methods and used documentation media types. Thirdly, it provided four general observations regarding NUI documentation-related tools and languages based on a review of a rich set of tools and languages found in the literature. Fourthly, it provided observations to highlight the designers' use of availability of adequate documentation tools, documentation standards, and the regularity of documentation habits. Fifthly, it presented observations to reveal the match and mismatch between the NUI designers' documentation practices and users' preferred learning practices.

This study reveals that good documentation practices are rare and largely compromised due to the lack of adequate documentation tools, absence of documentation standards, and irregularity of documentation habits. Moreover, the investigation highlights the impact of documentation practices on the users' learning practices, especially by exposing the existing mismatch gaps between preferred and available documentation methods and materials for users. Moreover, reviewing the existing documentation-related tools reveals four general observations including the lack of widely adopted tools by NUI designers, the absences of dedicated NUI documentation tools, the lack of end-user support, and the lack of support and considerations of body movements and postures as part of the interaction descriptions (if at all found). Furthermore, the review reveals that there is a lack of formalized languages and notations of generic motion. In fact, most of the discussed studies cited in this section neither target NI documentation per se nor provide a general methodology. Hence, the chapter strongly indicates that the issues of NUI documentation remains a challenging and unresting task for the HCI community. Hence, the creation of a collective long lasting *interaction heritage* remains unachievable and optimal user learning habits remain unsatisfied and weakly considered. In addition, the investigation aims to trigger a community-scale discussion to consider documentation as an important design measure for successful preservation, dissemination, and sharing of interaction techniques.

Current documentation and dissemination practices greatly challenge the realization of the *Interaction Ensemble*. The absence of standard documentation and dissemination strategies results into hindering the automatic and large scale deployment of interactions in ambient spaces. *Interaction Ensembles* can only be realized if interactive ambient systems, such as STAGE, have adequate information about the interaction technique including the interactions' movement components, the user context (physical abilities, disabilities, etc.), and interaction context (interaction primitives, etc.).

7.2. Future work

The work covered in this dissertation on *Interaction Ensembles* is only a starting point for researching and investigating this concept. While the presented ideas and the conceptual design provide good ground and foundation for adopting, adapting, and sharing NI in ambient spaces, they cannot be considered yet exclusive or comprehensive. Hence, the extensibility factor is carefully considered in most of this research contribution. Researching this topic is challenged by the limited understanding of design space of interactions in ambient spaces. Currently, related concepts, such as Interaction Primitives and Interaction Tasks, are neither well understood nor adequately validated in the context of ambient systems. For instance, interaction primitives are not adequately comprehensive to cover all interaction capabilities and can be mostly extended. Moreover, the interaction tasks are adopted mainly from GUI paradigm and more research is required to be investigated in the context of NUI paradigm.

The presented implementations in various parts of this thesis are targeted mainly to validate the feasibility and the scope of the conceptual design. Hence, they can be only considered as a proof-of-concept and only feature a subset of the proposed vision of this thesis. The implementation of the STAGE framework, based on handheld devices, is just one possible strategy, but ambient spaces offer more options and a wider range of possibilities. Hence, a more comprehensive implementation and evaluation should be inevitably carried out and continued.

In the presented implementation, a simple plugin repository solution was introduced. Nonetheless, designing adequate plugin repositories for runtime deployment of interactions remains an open research question. The design should consider the tension between private, public, and hybrid accessible and owned repositories. Other emerging issues also include repository management, security and privacy, and distributed plugin repositories.

The implementation of the movement profiles based on Labanotation covers only the structural subparts of Labanotation. Extensions of the current scheme can be appreciated to widen the movement description to include feeling and emotion descriptions. One interesting research area is the creation and generation of physical ability and movement profiles. With existing context acquisition frameworks, automatic building and generation of physical abilities and disabilities profiles can be of great help for interactive ambient systems. This automatic profiling benefits greatly from increasing deployment of sensor capabilities, emerging social and personal health services, etc. Moreover, aggregating and reasoning about the user's interaction behaviour may be used in automated profiling processes as well. The diversity of the user population and the wide range of physical assessment tests adds to the complexity and challenges of this research field.

Furthermore, the implementation of the Interaction Editor presents only the initial steps for a more comprehensive and dedicated editor for interactions. For instance, investigating the generation of an animated 3D human model based on interaction documentation and the integration with the editor may aid the design process of movement profiles greatly. Moreover, the editor should be extended to model ability and interaction profiles as well.

Moreover, improved and fully featured dedicated technical performance tests and evaluations should be specifically designed for this novel category of interactive systems. Another aspect to consider is the user centric evaluation of ensemble-driven interactive systems, which is very challenging due to the lack of standard evaluation measures. This requires a very deep investigation and novel evaluation methodologies to be designed and created by the HCI community. This research aspect is greatly challenged by the wide range of interaction techniques available, open-ended future innovation in the area of interaction techniques, and the open possibilities of NI ensembles.

The NUI documentation practices study conducted in this dissertation provides very important novel insights and opens various open questions. Nonetheless, the presented study is limited in scope and can be only considered as limited snapshot on existing practices. Further studies in this field are required for more extensive and complete representative results. Extending the investigation to study the differences and similarities between NUI documentation in academic and commercial settings (as in multitouch- and motion-based application market initiatives) can be of great use.

The adoption of ensemble-enabled systems requires inevitable changes within various interaction design and manufacturing processes. Soft-wiring approaches in the design and implementation of interaction techniques, devices, and applications should be applied. To achieve this vision, new design and development requirements, guidelines, and tools are required. Moreover, new design and implementation habits and practices should be enforced.

Finally, ensemble-enabled systems inherently challenge the learning and remembering of interaction techniques and ensembles. Principally, new demands and requirements for suitable interaction documentation emerge. Documentation for interaction ensembles should be dynamic and created in-situ. The most challenging aspects in this field are related to the composition and fusion of documentation from multiple heterogeneous multimodal documentation resources. Moreover, the adaptation of documentation according to the users' needs, abilities, and available media rendering resources is yet another important and open research field.

A

Author's personal information

The author's personal information is omitted in the digital version of the thesis. The information can be found however in the printed version of the thesis. The page count was retained for consistency between the digital and printed versions of the thesis.

Page **1** from 5

The author's personal information is omitted in the digital version of the thesis. The information can be found however in the printed version of the thesis. The page count was retained for consistency between the digital and printed versions of the thesis.

Page **2** from 5

The author’s personal information is omitted in the digital version of the thesis. The information can be found however in the printed version of the thesis. The page count was retained for consistency between the digital and printed versions of the thesis.

The author's personal information is omitted in the digital version of the thesis. The information can be found however in the printed version of the thesis. The page count was retained for consistency between the digital and printed versions of the thesis.

Page **4** from 5

The author's personal information is omitted in the digital version of the thesis. The information can be found however in the printed version of the thesis. The page count was retained for consistency between the digital and printed versions of the thesis.

B
Author's publications

- **Altakrouri, B.** and Schrader, A. Dynamic Interaction Plugins Deployment in Ambient Spaces. To appear in 5th International Conference on Human-Centered Software Engineering 2014 (HCSE'2014), September 16 - 18 2014, Paderborn, Germany.

- **Altakrouri, B.** and Schrader, A. Describing movements for motion gestures. In 1st International Workshop on Engineering Gestures for Multimodal Interfaces (EGMI 2014) at the sixth ACM SIGCHI Symposium on Engineering Interactive Computing Systems (EICS'14), June 17 2014, Rome, Italy.

- Carlson, D., **Altakrouri, B.**, Schrader, A. An Ad-hoc Smart Gateway Platform for the Web of Things. In Proceedings of the IEEE International Conference on Internet of Things (iThings 2013); Beijing, China, August 2013. [iThings 2013 Best Paper Award]

- **Altakrouri, B.**, Carlson, D., and Schrader, A. Sharing kinetic interactions for mobile devices. In Design, User Experience, and Usability. User Experience in Novel Technological Environments, A. Marcus, Ed., vol. 8014 of Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2013, pp. 327 - 336.

- **Altakrouri, B.**, Gröschner, J., and Schrader, A. Documenting natural interactions. In CHI'13 Extended Abstracts on Human Factors in Computing Systems (New York, NY, USA, 2013), CHI EA'13, ACM, pp. 1173 - 1178.

- Ruge, L., **Altakrouri, B.**, Schrader, A. SoundOfTheCity - Continuous Noise Monitoring for a Healthy City. 5th International Workshop on Smart Environments and Ambient Intelligence (SENAmI 2013) at IEEE International Conference on Pervasive Computing and Communication (PerCom 2013), San Diego, California, , USA, March 18-22 2013.

- Carlson, D., **Altakrouri, B.**, Schrader, A. AmbientWeb: Bridging the Web's Cyber-physical Gap. In Proceedings of the 3rd International Conference on the Internet of Things (IoT2012), 2012. [IoT2012 Challenge Winner] .

- **Altakrouri, B.** and Schrader, A. Towards dynamic natural interaction ensembles. In Fourth International Workshop on Physicality (Physicality 2012) co-located with British HCI 2012 conference (Birmingham, UK, 09 2012), A. D. Devina Ramduny-Ellis and S. Gill, Eds.

- **Altakrouri, B.**, Kortuem, G., Grünerbl, A., Lukowicz, P. The Benefit of Activity Recognition for Mobile Phone Based Nursing Documentation: Wizard-of-Oz Study. Accepted at ISWC'10, the fourteenth annual IEEE International Symposium on Wearable Computers, Oct. 10th 13th 2010, COEX, Seoul, South Korea. (http://www.iswc.net/iswc10/inc_html/).

- Kawsar, F., Kortuem, G., **Altakrouri, B.** Supporting Interaction with the Internet of Things across Objects, Time and Space. Accepted at the Internet of Things 2010 Conference (IoT-2010), Nov 29 - Dec 1 2010, Tokyo, Japan. (http://www.iot2010.org/)

- Kawsar, F., Kortuem, G., **Altakrouri, B.** Designing Pervasive Interactions for Ambient Guidance with Situated Flows. Fourth International Workshop on Human Aspects in Ambient Intelligence, in collaboration with the 2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT 2010), Toronto, Canada, August31- September 3, 2010.

- **Altakrouri, B.**, Kawsar, F., Kortuem, G. Spin&Swing : Spatial Interaction with Orientation Aware Devices. Demo paper in the Eighth International Conference on Pervasive Computing (Pervasive 2010), Helsinki, Finland, 17 - 20 May 2010. (http://www.pervasive2010.org/).

- Kortuem, G., Kawsar, F., **Altakrouri, B.** Flow Driven Ambient Guidance. Eighth Annual IEEE International Conference on Pervasive Computing and Communications (PerCom 2010), Mannheim, Germany March 29 - April 2, 2010. (http://www.percom.org/).

- **Altakrouri, B.** Flow-driven Interactions for Adaptive Pervasive Applications. Abstract at Mobile and Embedded Interactive Systems (MEIS 2009), Informatik 2009, Luebeck, Germany, September 28 - October 02, 2009. (http://www.hcilab.org/events/meis09/).

- **Altakrouri, B.**, Schrader, A. Ambient Media Services in Intelligent Environments. In: Callaghan, V.; Kameas, A.; Reyes, A.; Royo, D., Weber, M. (eds.): Proceedings of the 5th International Conference on Intelligent Environments. IE09. Special Session on Awareness Systems. Volume 2 Ambient Intelligence and Smart Environments, July 20-21, 2009, Barcelona, Spain. IOS Press, Amsterdam, 2009. ISBN 978-1-60750-034-6, pp. 75-82. (IOS Press, Amsterdam.)

- **Altakrouri, B.**, Detken, K., Martinez, C., Oja, M., Stein, S., Zhu, L., Schrader, A. Mobile HolstenTour: Contextualized Multimedia Museum Guide. Accepted and published at International Workshop on Ubiquitous Web Applications at the 10th International Conference on Information Integration and Web-based Applications & Services (ACM iiWAS 2008) November 24-26, 2008, Linz, Austria. (http://www.iiwas.org/conferences/iiwas2008).

- **Altakrouri, B.**, Bustamante, L., Canonico, A., Janiszewski, M., Nikolova, E., Talmantas, M., Schrader, A. Pictory - A Hybrid Augmented Table-Top Game using Image Search Engines. Accepted and published at the 4th International Symposium on Pervasive Gaming applications (PERGAMES'2007), June 11-12, 2007,Salzburg, Austria. Shaker Verlag, Aachen 2007, . (www.pergames.de).

- **Altakrouri, B.**, Canonico, A., Gongora, L., Janiszewski, M., Toader, C., and Schrader, A. eyeJOT - A Ubiquitous Context-aware Campus Information System. Published at the Second International Conference on Pervasive Computing and Applications (ICPCA07), IEEE press, July 26-27, 2007, Birmingham, UK (http://www.icpca.uce.ac.uk/).

- Hasebrook, J., Janiszewski, M., Mansilla, W., Schmidts, U., **Altakrouri, B.** PortNet Navigator:Internet-Based Visual Planning Tool for Maritime Logistics. Published at The 3rd International Symposium on Management, Engineering and Informatics: MEI 2007. July 8-11, 2007, Orlando, Florida, USA. (http://www.iiis-cyber.org/wmsci2007/).

- de Jong, T., **Altakrouri, B.**, Specht, M., Koper, R. (November, 2006). Campus Memories: Learning with Contextualised Blogging. Paper presented at the TENCompetence workshop, Manchester, 11th 12th January 2007.

C
Design guidelines for ambient interfaces

Characteristic	Explanation
Effective	Quality in terms of how good the ambient interfaces do what they are supposed to do.
Efficient	Way an ambient interface supports users in carrying out their tasks.
Safe	Protection of the user from dangerous and undesirable situations.
Good utility	Right kind of functionality so that users can do what they want and need to do.
Easy to learn and remember	Ease for the users to learn a system and to remember how to interact with the system.
Visible functionality	Clear communication to the user at any time which choice she has and what the system is expecting from her.
Provide adequate feedback	Information to the users to tell her that her input was received and analyzed properly, and that the corresponding actions have been or will be performed.
Provide constraints	System awareness of the user's current situation and possible next steps and appropriate actions of a user.
Adequate mapping	Mapping between controls and their effects should be adequate.
Consistent functionality	Similar operations and similar control elements should be used for achieving similar tasks.
Adequate for the target domain	Adequacy for the target domain such as the environment, in which the ambient interfaces are installed; the users, who will use the system; and the tasks that will be performed on the ambient interfaces.
Participatory	Stimulation of users to contribute to the design of the ambient interfaces at very early stages.

Table C.1.: Design guidelines for ambient interfaces (adopted from [65])

D
ICF related information

Box 6: The generic qualifier and an example of an ICF-code

ICF codes require the use of one or more qualifiers which denote the magnitude or severity of the problem in question. The problem refers to an impairment, limitation, restriction, or barrier when used in combination with b, s, d or e codes, respectively. Qualifiers are coded as one or more numbers after a decimal point.

xxx.0	NO problem	(none, absent, negligible, ...)	0-4%
xxx.1	MILD problem	(slight, low, ...)	5-24%
xxx.2	MODERATE problem	(medium, fair, ...)	25-49%
xxx.3	SEVERE problem	(high, extreme, ...)	50-95%
xxx.4	COMPLETE problem	(total, ...)	96-100%
xxx.8	not specified		
xxx.9	not applicable		

The letters b, s, d, and e represent the different components and are followed by a numeric code that starts with the chapter number (one digit), followed by the second level (two digits), as well as third and fourth levels (one extra digit each). For example, the following codes indicate a 'mild' problem in each case.

b2.1	Sensory functions and pain	(first-level item)
b210.1	Seeing functions	(second-level item)
b2102.1	Quality of vision	(third-level item)
b21022.1	Contrast sensitivity	(fourth-level item)

WHO 2001

Figure D.1.: The generic qualifier and an example of an ICF-code (from [130], p.26)

Components	First qualifier	Second qualifier
Body Functions (b)	Generic qualifier with the negative scale used to indicate the extent or magnitude of an impairment Example: b167.3 to indicate a severe impairment in specific mental functions of language	None
Body Structures (s)	Generic qualifier with the negative scale used to indicate the extent or magnitude of an impairment Example: s730.3 to indicate a severe impairment of the upper extremity	Used to indicate the nature of the change in the respective body structure: 0 no change in structure 1 total absence 2 partial absence 3 additional part 4 aberrant dimensions 5 discontinuity 6 deviating position 7 qualitative changes in structure, including accumulation of fluid 8 not specified 9 not applicable Example: s730.32 to indicate the partial absence of the upper extremity
Activities and Participation (d)	Performance Generic qualifier Problem in the person's current environment Example: d5101.1_ to indicate mild difficulty with bathing the whole body with the use of assistive devices that are available to the person in his or her current environment	Capacity Generic qualifier Limitation without assistance Example: d5101._2 to indicate moderate difficulty with bathing the whole body; implies that there is moderate difficulty without the use of assistive devices or personal help
Environmental Factors (e)	Generic qualifier, with negative and positive scale, to denote extent of barriers and facilitators respectively Example: e130.2 to indicate that products for education are a moderate barrier. Conversely, e130+2 would indicate that products for education are a moderate facilitator	None

Table D.1.: ICF Qualifiers (from [129], p.24)

E
Ability profile XML scheme

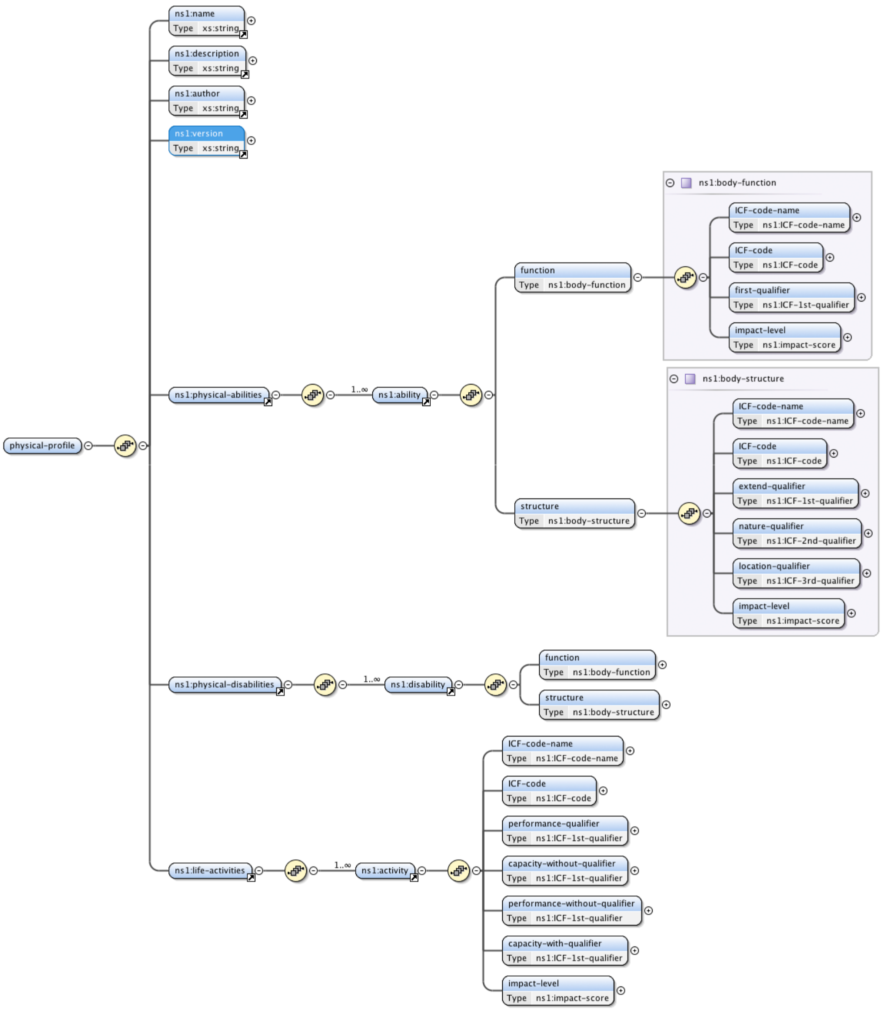
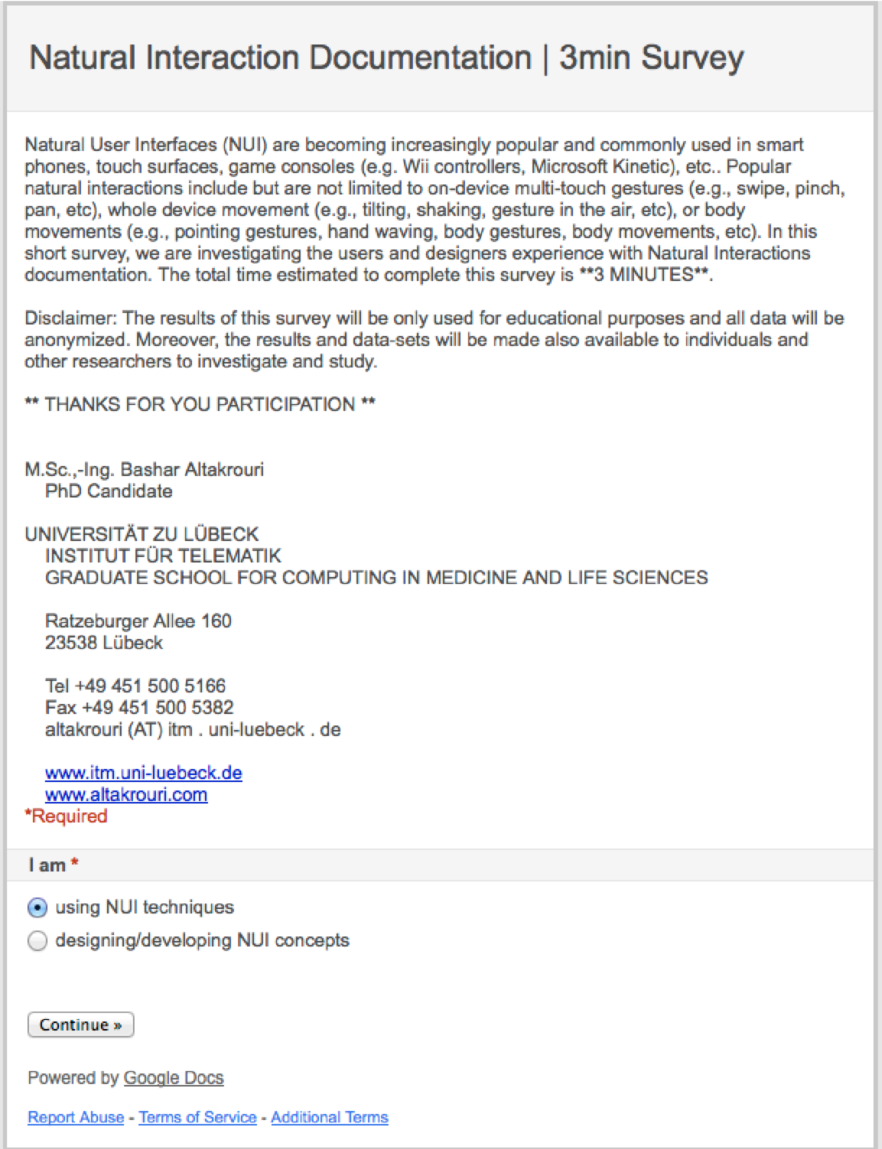


Figure E.1.: Ability profile XML scheme - full overview

F
Natural interaction documentation | 3min survey



Natural Interaction Documentation | 3min Survey

Natural User Interfaces (NUI) are becoming increasingly popular and commonly used in smart phones, touch surfaces, game consoles (e.g. Wii controllers, Microsoft Kinetic), etc.. Popular natural interactions include but are not limited to on-device multi-touch gestures (e.g., swipe, pinch, pan, etc), whole device movement (e.g., tilting, shaking, gesture in the air, etc), or body movements (e.g., pointing gestures, hand waving, body gestures, body movements, etc). In this short survey, we are investigating the users and designers experience with Natural Interactions documentation. The total time estimated to complete this survey is **3 MINUTES**.

Disclaimer: The results of this survey will be only used for educational purposes and all data will be anonymized. Moreover, the results and data-sets will be made also available to individuals and other researchers to investigate and study.

** THANKS FOR YOU PARTICIPATION **

M.Sc.,-Ing. Bashar Altakrouri
PhD Candidate

UNIVERSITÄT ZU LÜBECK
INSTITUT FÜR TELEMATIK
GRADUATE SCHOOL FOR COMPUTING IN MEDICINE AND LIFE SCIENCES

Ratzeburger Allee 160
23538 Lübeck

Tel +49 451 500 5166
Fax +49 451 500 5382
altakrouri (AT) itm . uni-luebeck . de

www.itm.uni-luebeck.de
www.altakrouri.com

*Required

I am *

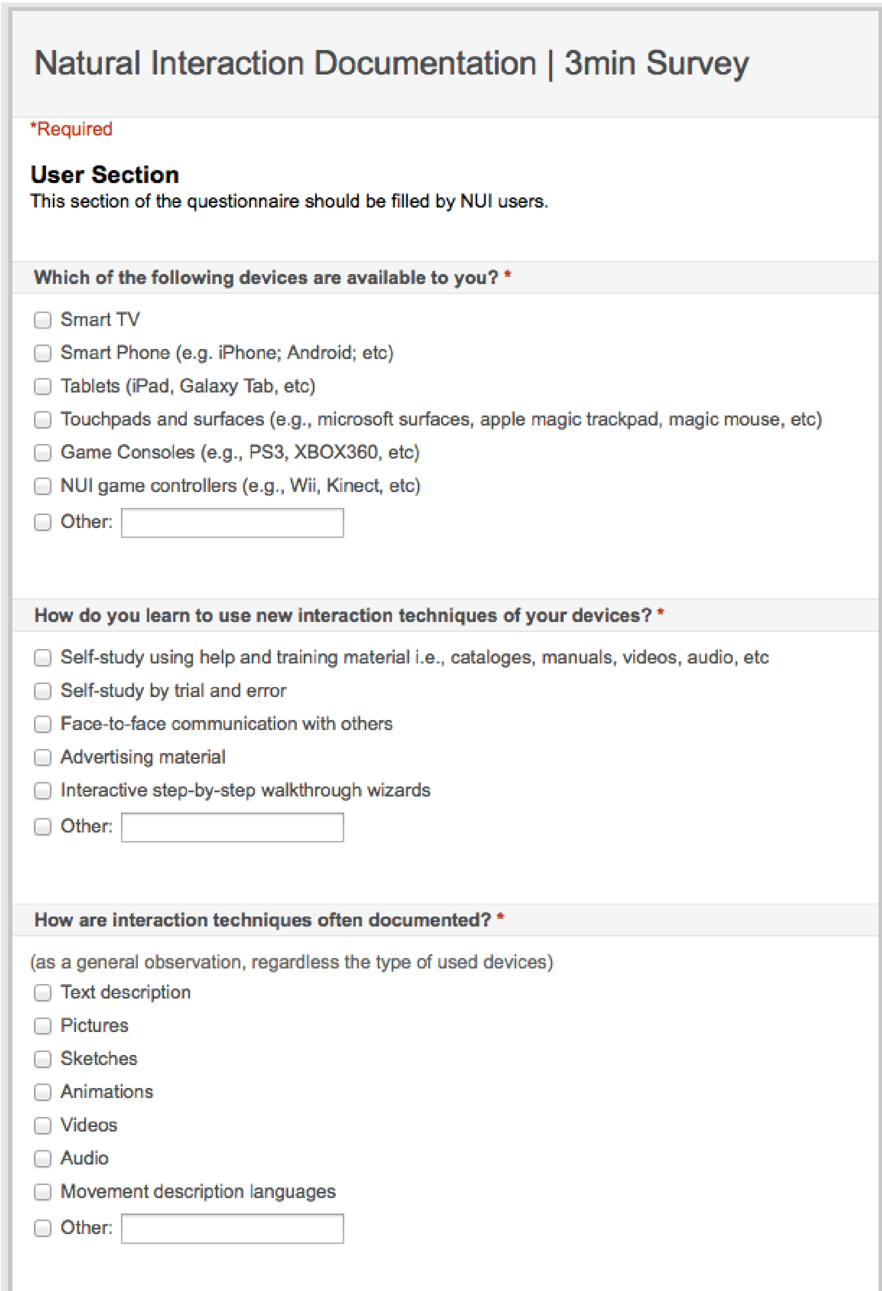
- (•) using NUI techniques
- ( ) designing/developing NUI concepts

Continue »

Powered by Google Docs

Report Abuse - Terms of Service - Additional Terms

Figure F.1.: Survey introduction



Natural Interaction Documentation | 3min Survey

*Required

User Section
This section of the questionnaire should be filled by NUI users.

Which of the following devices are available to you? *

- ☐ Smart TV
- ☐ Smart Phone (e.g. iPhone; Android; etc)
- ☐ Tablets (iPad, Galaxy Tab, etc)
- ☐ Touchpads and surfaces (e.g., microsoft surfaces, apple magic trackpad, magic mouse, etc)
- ☐ Game Consoles (e.g., PS3, XBOX360, etc)
- ☐ NUI game controllers (e.g., Wii, Kinect, etc)
- ☐ Other: ____

How do you learn to use new interaction techniques of your devices? *

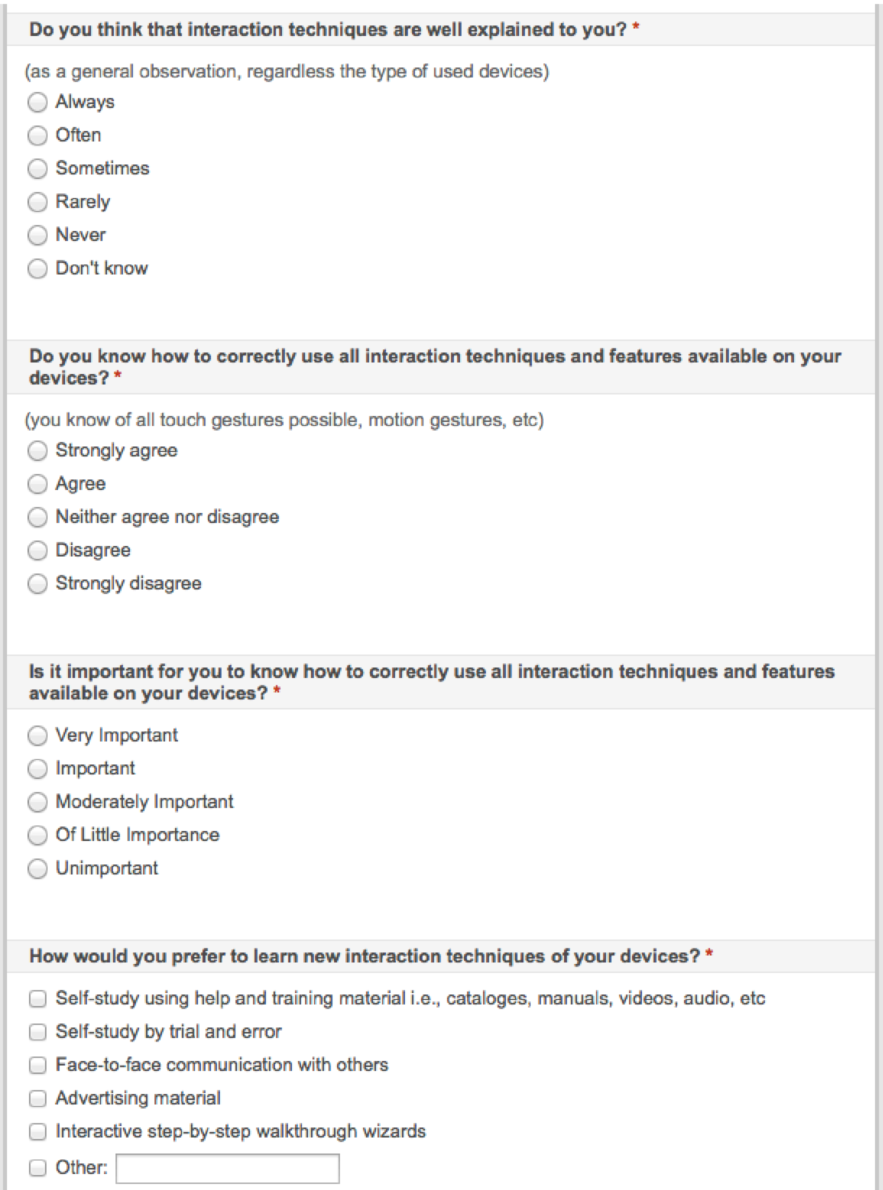
- ☐ Self-study using help and training material i.e., cataloges, manuals, videos, audio, etc
- ☐ Self-study by trial and error
- ☐ Face-to-face communication with others
- ☐ Advertising material
- ☐ Interactive step-by-step walkthrough wizards
- ☐ Other: ____

How are interaction techniques often documented? *

(as a general observation, regardless the type of used devices)

- ☐ Text description
- ☐ Pictures
- ☐ Sketches
- ☐ Animations
- ☐ Videos
- ☐ Audio
- ☐ Movement description languages
- ☐ Other: ____

Figure F.2.: NUI user section questions 1 - 3



Do you think that interaction techniques are well explained to you? *

(as a general observation, regardless the type of used devices)

- ◯ Always
- ◯ Often
- ◯ Sometimes
- ◯ Rarely
- ◯ Never
- ◯ Don't know

Do you know how to correctly use all interaction techniques and features available on your devices? *

(you know of all touch gestures possible, motion gestures, etc)

- ◯ Strongly agree
- ◯ Agree
- ◯ Neither agree nor disagree
- ◯ Disagree
- ◯ Strongly disagree

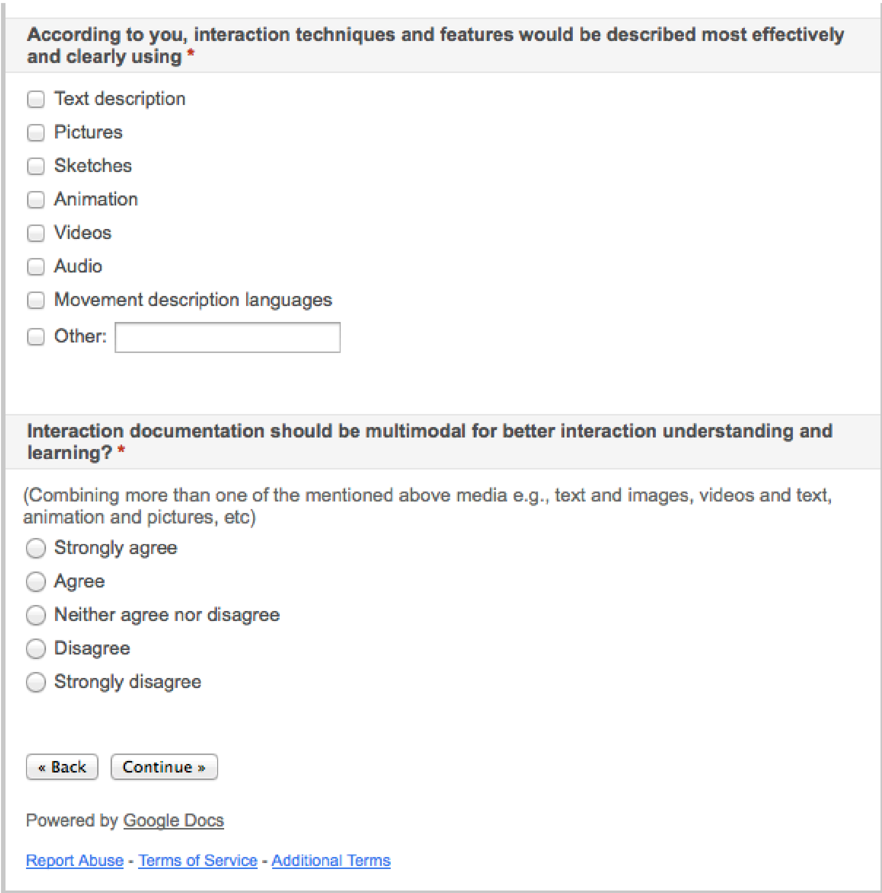
Is it important for you to know how to correctly use all interaction techniques and features available on your devices? *

- ◯ Very Important
- ◯ Important
- ◯ Moderately Important
- ◯ Of Little Importance
- ◯ Unimportant

How would you prefer to learn new interaction techniques of your devices? *

- ☐ Self-study using help and training material i.e., cataloges, manuals, videos, audio, etc
- ☐ Self-study by trial and error
- ☐ Face-to-face communication with others
- ☐ Advertising material
- ☐ Interactive step-by-step walkthrough wizards
- ☐ Other: ____

Figure F.3.: NUI user section questions 4 - 7



According to you, interaction techniques and features would be described most effectively and clearly using *

- [ ] Text description
- [ ] Pictures
- [ ] Sketches
- [ ] Animation
- [ ] Videos
- [ ] Audio
- [ ] Movement description languages
- [ ] Other:

Interaction documentation should be multimodal for better interaction understanding and learning? *

(Combining more than one of the mentioned above media e.g., text and images, videos and text, animation and pictures, etc)

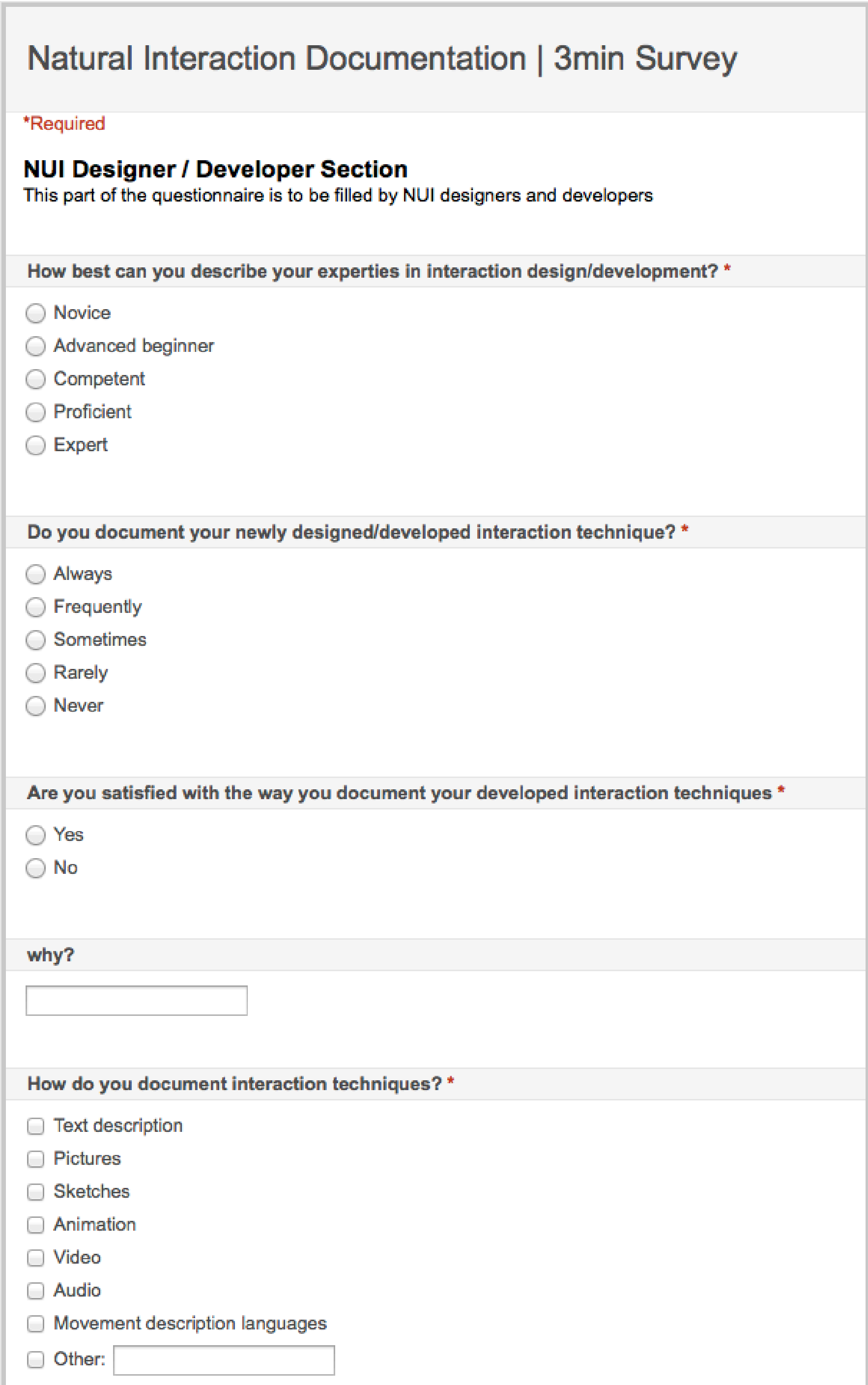
- Strongly agree
- Agree
- Neither agree nor disagree
- Disagree
- Strongly disagree

« Back | Continue »

Powered by Google Docs

Report Abuse - Terms of Service - Additional Terms

Figure F.4.: NUI user section questions 8 - 9



Natural Interaction Documentation | 3min Survey

*Required

NUI Designer / Developer Section

This part of the questionnaire is to be filled by NUI designers and developers

How best can you describe your experties in interaction design/development? *

- ◯ Novice
- ◯ Advanced beginner
- ◯ Competent
- ◯ Proficient
- ◯ Expert

Do you document your newly designed/developed interaction technique? *

- ◯ Always
- ◯ Frequently
- ◯ Sometimes
- ◯ Rarely
- ◯ Never

Are you satisfied with the way you document your developed interaction techniques *

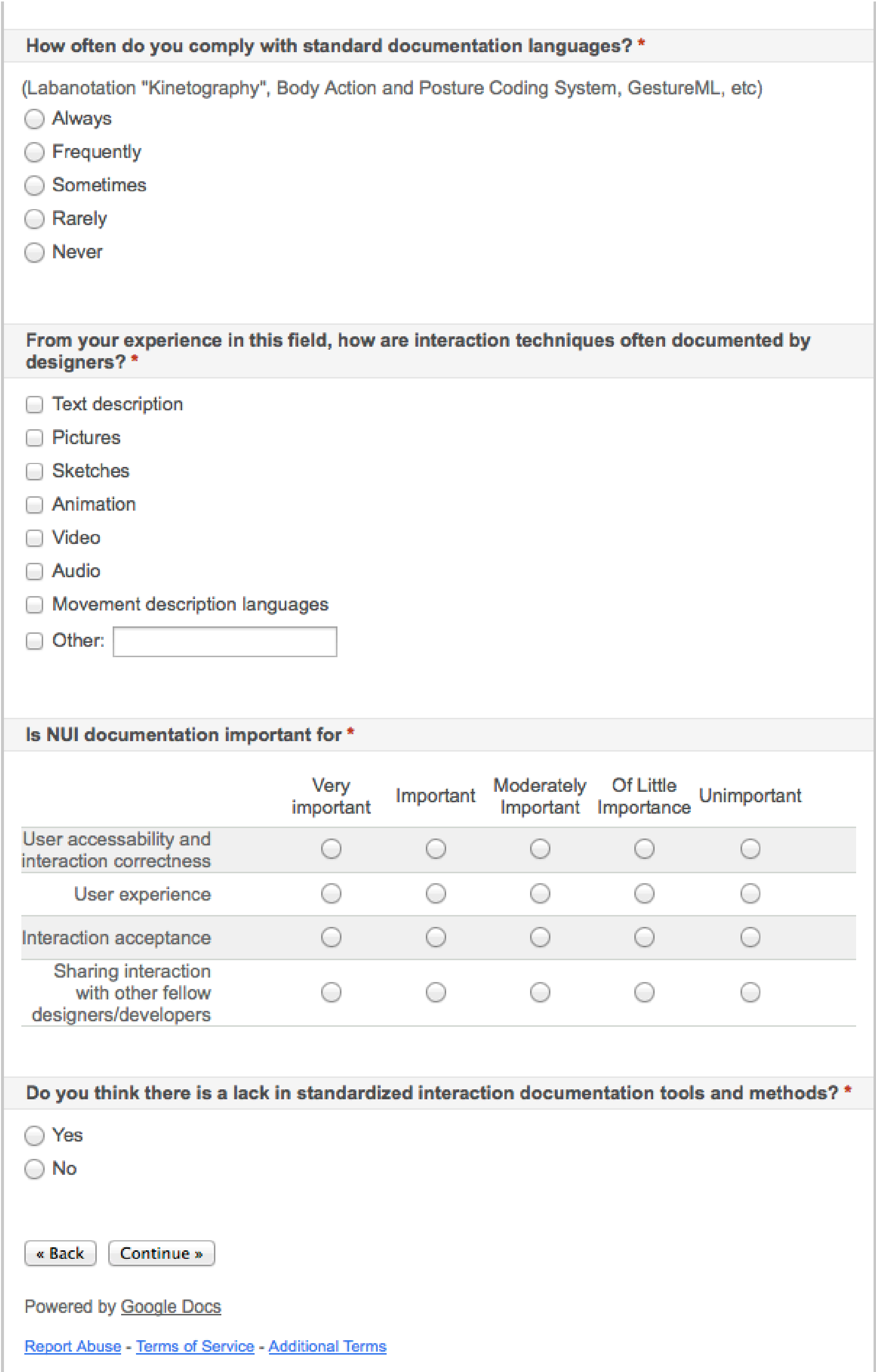
- ◯ Yes
- ◯ No

why?

How do you document interaction techniques? *

- ☐ Text description
- ☐ Pictures
- ☐ Sketches
- ☐ Animation
- ☐ Video
- ☐ Audio
- ☐ Movement description languages
- ☐ Other:

Figure F.5.: NUI designer section questions 1 - 4



How often do you comply with standard documentation languages? *

(Labanotation "Kinetography", Body Action and Posture Coding System, GestureML, etc)

- Always
- Frequently
- Sometimes
- Rarely
- Never

From your experience in this field, how are interaction techniques often documented by designers? *

- Text description
- Pictures
- Sketches
- Animation
- Video
- Audio
- Movement description languages
- Other:

Is NUI documentation important for *

	Very important	Important	Moderately Important	Of Little Importance	Unimportant
User accessability and interaction correctness	○	○	○	○	○
User experience	○	○	○	○	○
Interaction acceptance	○	○	○	○	○
Sharing interaction with other fellow designers/developers	○	○	○	○	○

Do you think there is a lack in standardized interaction documentation tools and methods? *

- Yes
- No

« Back   Continue »

Powered by Google Docs

Report Abuse - Terms of Service - Additional Terms

Figure F.6.: NUI designer section questions 5 - 10

G
Interaction Editor screenshots

This appendix presents a number of magnified screenshots for the Interaction Editor. The same figures are presented in section 6.6.3 but with smaller size. This appendix is intended to improve the readability of those figures.

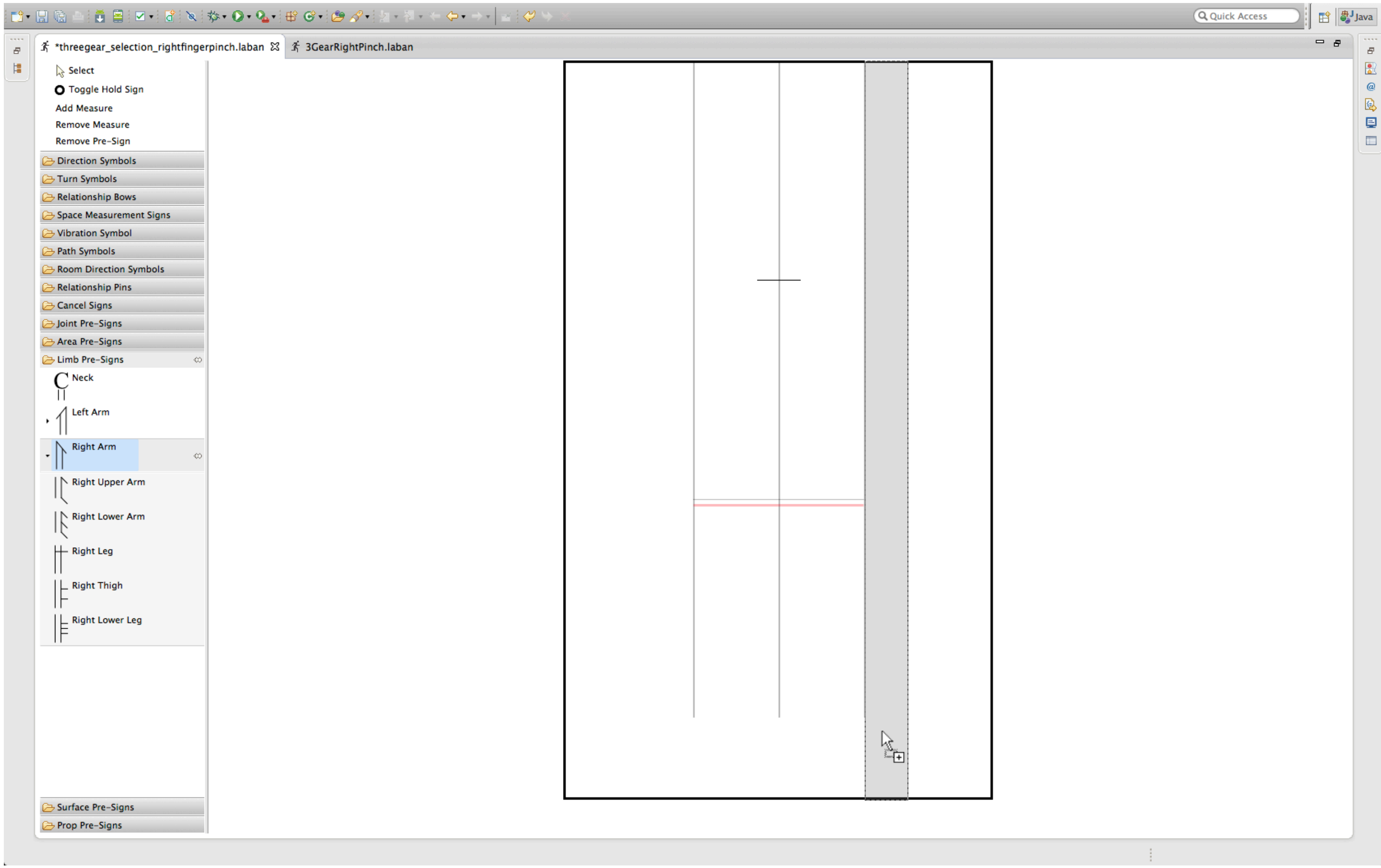


Figure G.1.: Interaction Editor: Adding a column definitions to the score

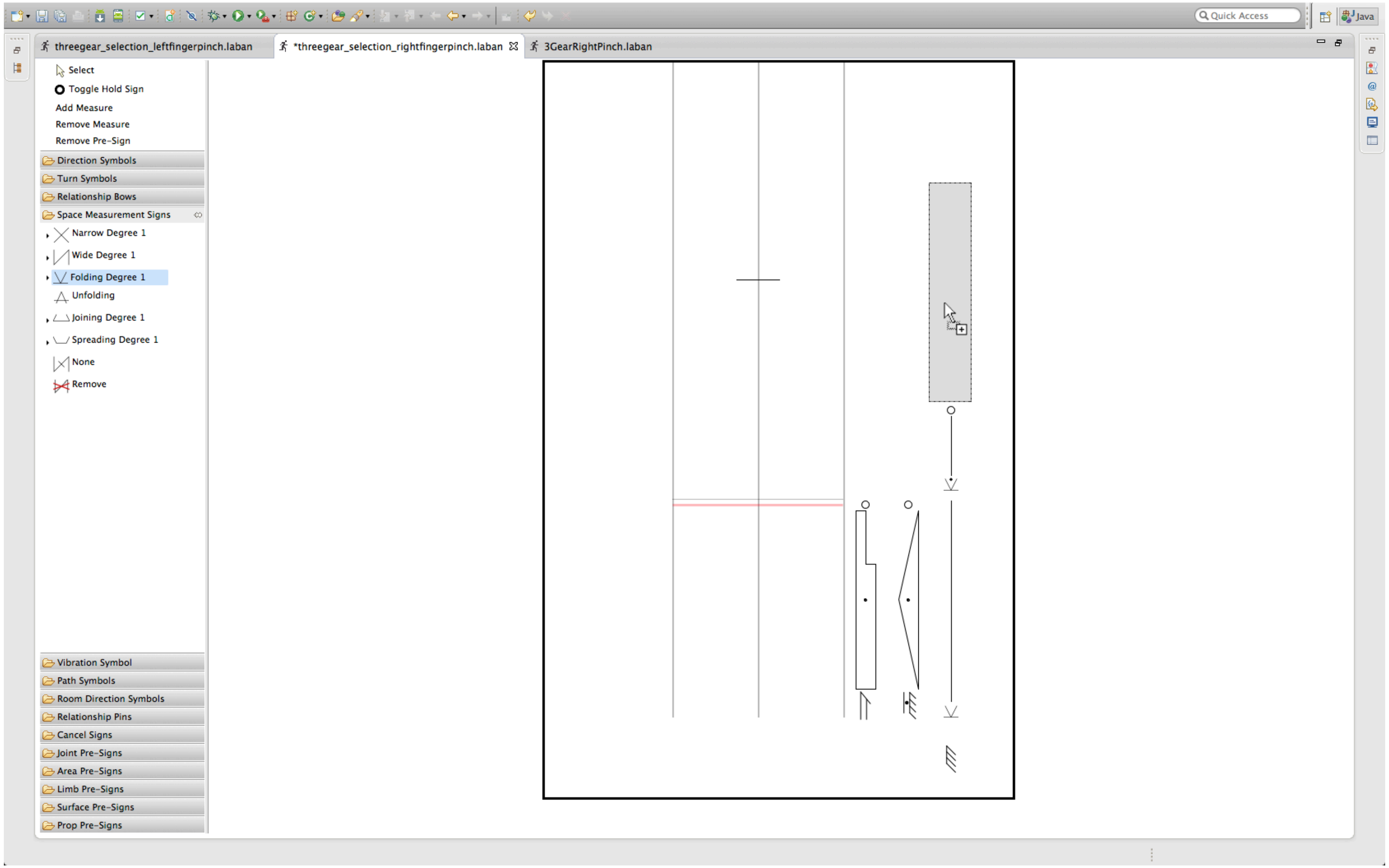


Figure G.2.: Interaction Editor: Adding a folding symbol to the score

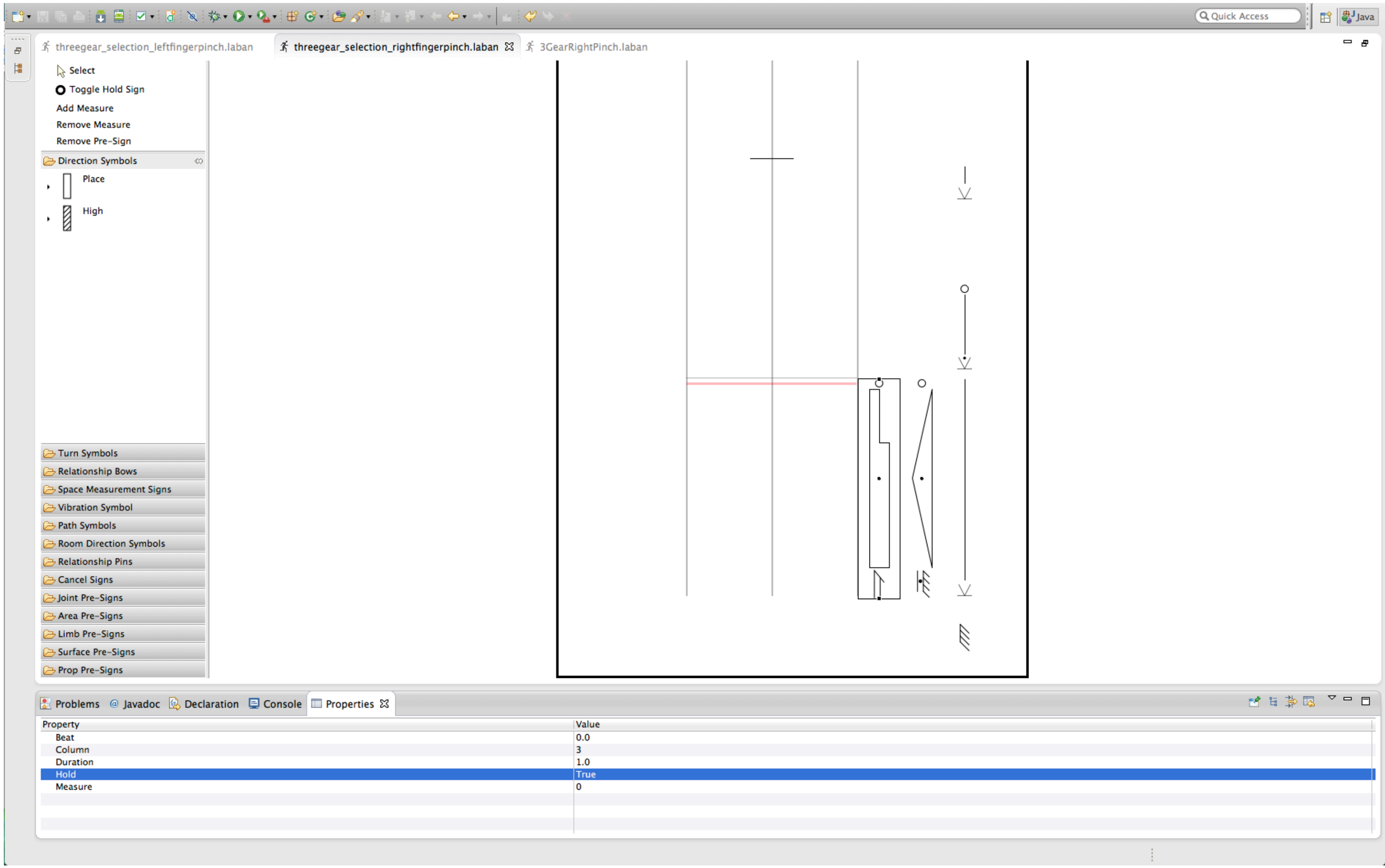


Figure G.3.: Interaction Editor: Changing the properties of a movement component

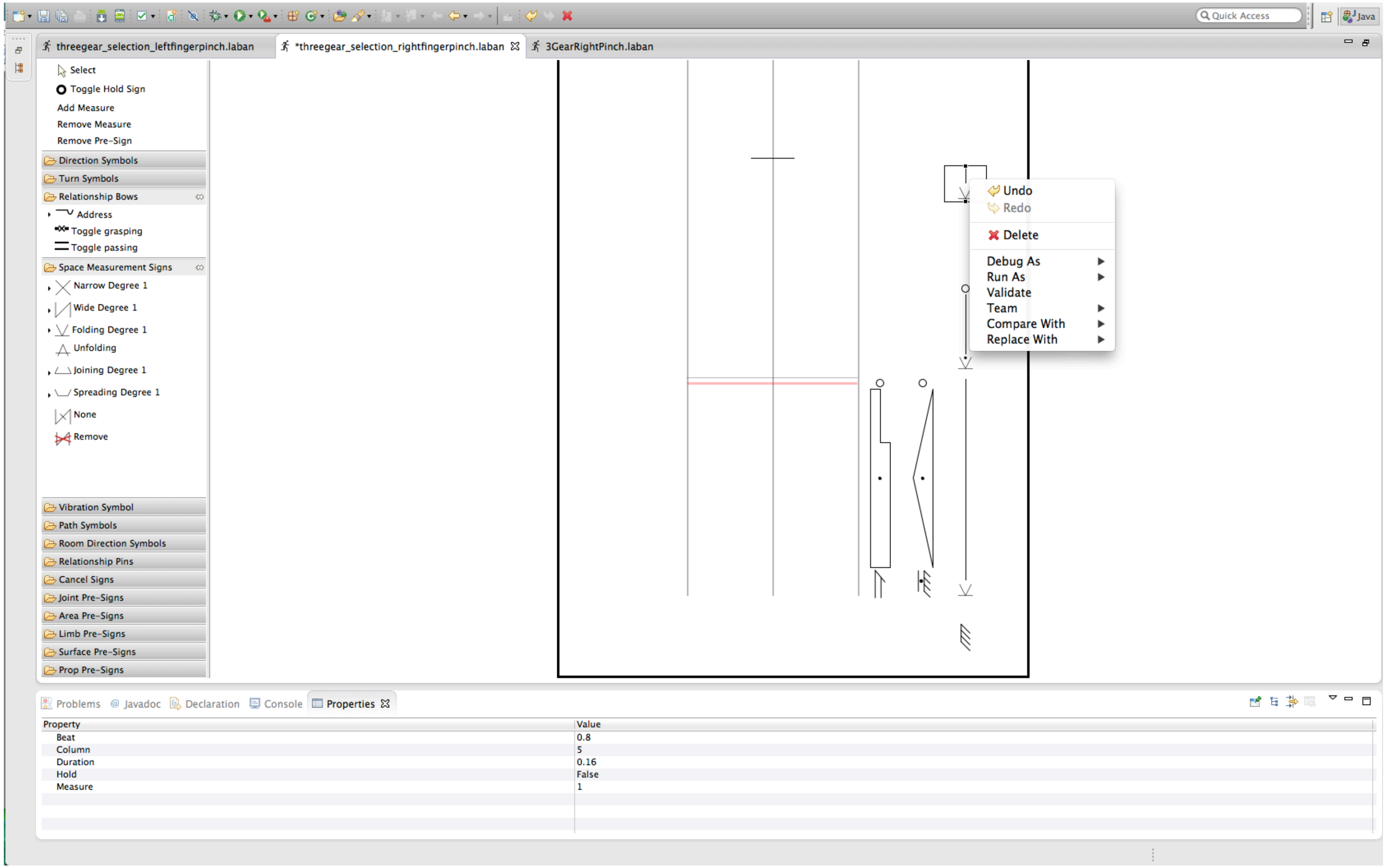


Figure G.4.: Interaction Editor: Deleting a movement component form the score

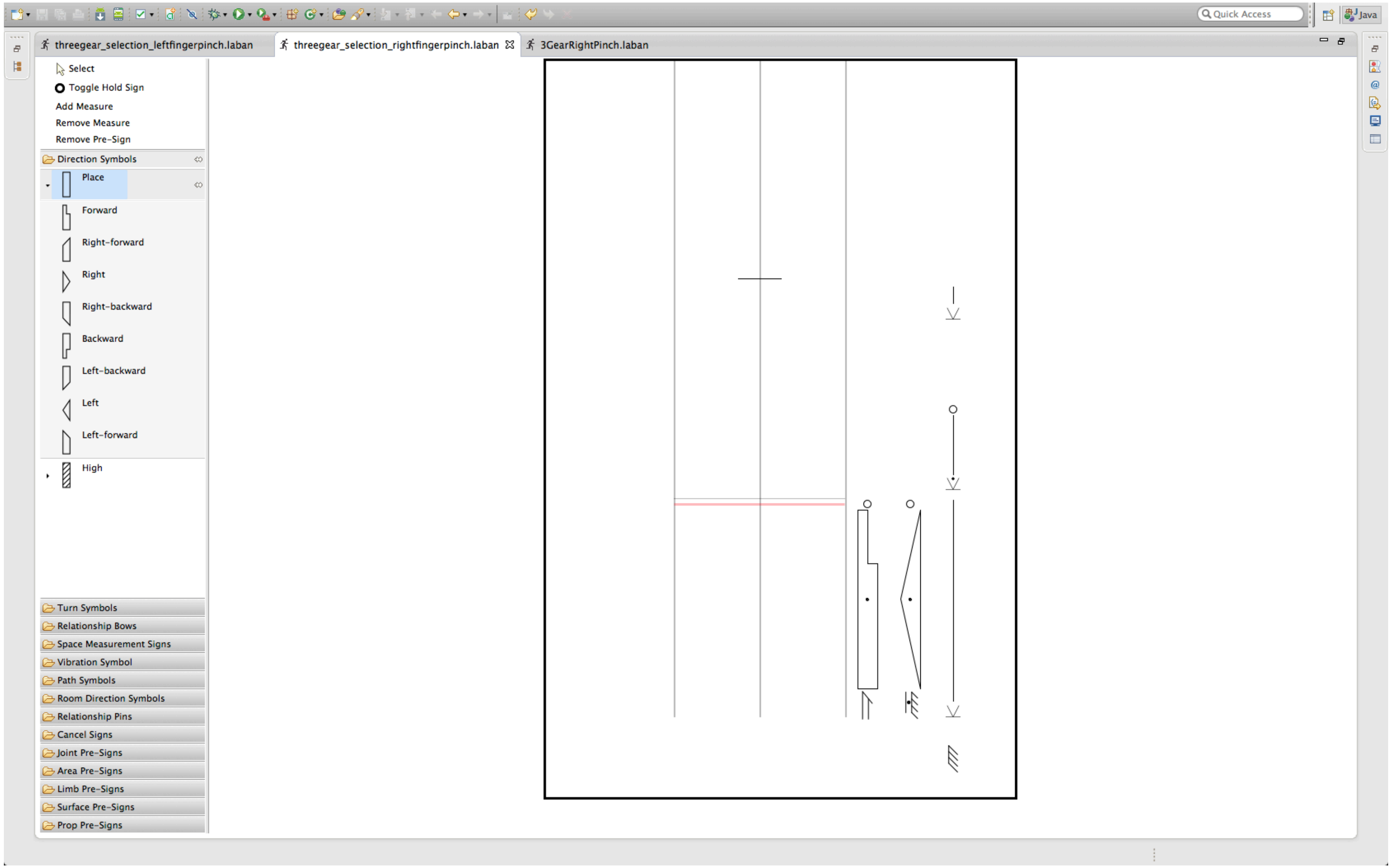


Figure G.5.: Interaction Editor: Complete Movement Profile in Labanotation visual representation

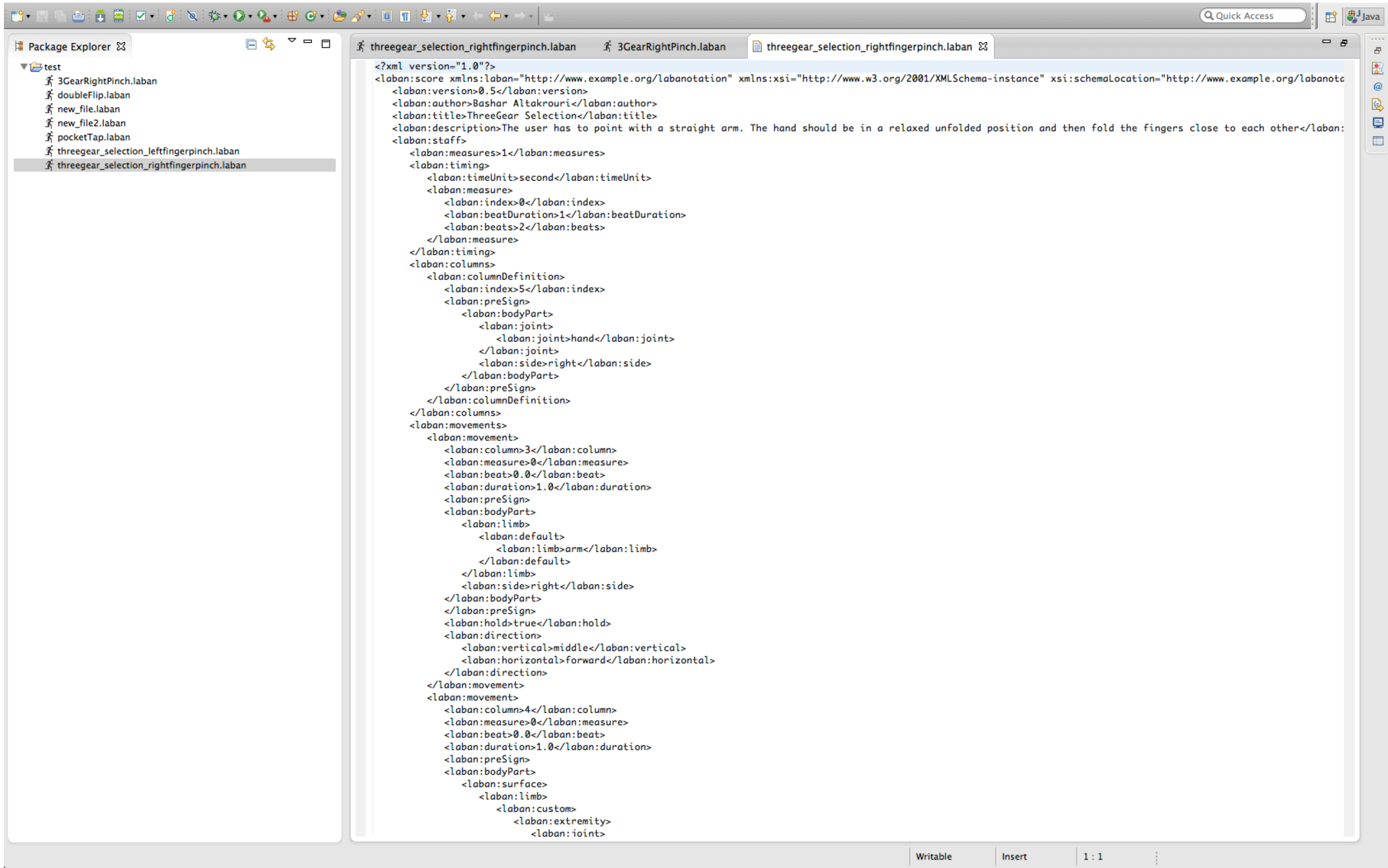


Figure G.6.: Interaction Editor: Complete Movement Profile in XML representation

Listing

List of Figures

List of Tables

List of listings

List of Acronyms

AAL Ambient Assisted Living

ACM Association for Computing Machinery

adb Android Debug Bridge

ADT Android Developer Tools

AmI Ambient Intelligence

API Application Programming Interface

CHI ACM Conference on Human Factors in Computing Systems

CPD Context Plugin Description

EICS ACM SIGCHI Symposium on Engineering Interactive Computing Systems

EU European Union

GEF Graphical Editing Framework

GDL Gesture Definition Language

GML Gesture Markup Language

GUI Graphical User Interface

HCI Human Computer Interaction

ICF International Classification of Functioning, Disability and Health

ICT Information and Communications Technology

IP Interaction Plugin

LMA Laban Movement Analysis

NI Natural Interaction

NUI Natural User Interface

OECD Organization for Economic Co-operation and Development

OSGi Open Services Gateway initiative

PC Personal Computer

POJO Plain Old Java Object

RFID Radio-frequency identification

SDK Software Development Kit

UIST ACM Symposium on User Interface Software and Technology

URI Uniform Resource Identifier

WHO World Health Organization

XML Extensible Markup Language

2D Two-dimensional Space

3D Three-dimensional Space

Bibliography

[1] Abowd, G. D., and Mynatt, E. D. Charting past, present, and future research in ubiquitous computing. *ACM Trans. Comput.-Hum. Interact. 7* (March 2000), 29–58.

[2] Aggarwal, J. K., and Cai, Q. Human motion analysis: A review. *Computer Vision and Image Understanding 73* (1999), 428–440.

[3] Alaoui, S. F., Caramiaux, B., Serrano, M., and Bevilacqua, F. Movement qualities as interaction modality. In *Proceedings of the Designing Interactive Systems Conference* (Newcastle, UK, 2012), DIS '12, ACM, pp. 761–769.

[4] Altakrouri, B., Canonico, A., Gongora, L., Janiszewski, M., Toader, C., and Schrader, A. eyeJOT - a ubiquitous context-aware campus information system. In *Second International Conference on Pervasive Computing and Applications (ICPCA07)* (Birmingham, UK, July 26-27 2007), IEEE press.

[5] Altakrouri, B., Carlson, D., and Schrader, A. Sharing kinetic interactions for mobile devices. In *Design, User Experience, and Usability. User Experience in Novel Technological Environments*, A. Marcus, Ed., vol. 8014 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2013, pp. 327–336.

[6] Altakrouri, B., Detken, K., Martinez, C., Oja, M.-K., Stein, S., Zhu, L., and Schrader, A. Mobile holstentour: contextualized multimedia museum guide. In *MoMM'2008 - The 6th International Conference on Advances in Mobile Computing and Multimedia* (Linz, Austria, 24 - 26 November 2008), ACM, pp. 460–463.

[7] Altakrouri, B., Gröschner, J., and Schrader, A. Documenting natural interactions. In *CHI '13 Extended Abstracts on Human Factors in Computing Systems* (New York, NY, USA, 2013), CHI EA '13, ACM, pp. 1173–1178.

[8] Altakrouri, B., Kawsar, F., and Kortuem, G. Spin&swing : Spatial interaction with orientation aware devices. In *The Eighth International Conference on Pervasive Computing (Pervasive 2010), Demo Section* (Helsinki, Finland, 17 - 20 May 2010).

[9] Altakrouri, B., and Schrader, A. Ambient media services in intelligent environments. In *Proceedings of the 5th International Conference on Intelligent Environments. IE09. Special Session on Awareness Systems* (Barcelona, Spain, July 20-21 2009), V. Callaghan, A. Kameas, A. Reyes, D. Royo, and M. e. Weber, Eds., vol. Volume 2 Ambient Intelligence and Smart Environments of *ISBN 978-1-60750-034-6*, IOS Press, Amsterdam, pp. 75–82.

[10] Altakrouri, B., and Schrader, A. Towards dynamic natural interaction ensembles. In *Fourth International Workshop on Physicality (Physicality 2012) co-located with British HCI 2012 conference* (Birmingham, UK, 09 2012), A. D. Devina Ramduny-Ellis and S. Gill, Eds.

[11] Altakrouri, B., and Schrader, A. Describing movements for motion gestures. In *1st International Workshop on Engineering Gestures for Multimodal Interfaces (EGMI 2014) at the sixth ACM SIGCHI Symposium on Engineering Interactive Computing Systems (EICS14)* (Rome, Italy, June 2014).

[12] Anthony, L., Kim, Y., and Findlater, L. Analyzing user-generated youtube videos to understand touchscreen use by people with motor impairments. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2013), CHI '13, ACM, pp. 1223–1232.

[13] Ashbrook, D., and Starner, T. Magic: a motion gesture design tool. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2010), CHI '10, ACM, pp. 2159–2168.

[14] Badler, N. I., and Smoliar, S. W. Digital representations of human movement. *ACM Comput. Surv. 11*, 1 (Mar. 1979), 19–38.

[15] Bailly, G., Vo, D.-B., Lecolinet, E., and Guiard, Y. Gesture-aware remote controls: guidelines and interaction technique. In *Proceedings of the 13th international conference on multimodal interfaces* (New York, NY, USA, 2011), ICMI '11, ACM, pp. 263–270.

[16] Ballagas, R., Ringel, M., Stone, M., and Borchers, J. istuff: A physical user interface toolkit for ubiquitous computing environments. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2003), CHI '03, ACM, pp. 537–544.

[17] Bartlett, J. F., and Bartlett, J. F. Rock 'n' scroll is here to stay. *Computer Graphics and Applications* (2000), 40–45.

[18] Becker, M., Werkman, E., Anastasopoulos, M., and Kleinberger, T. Approaching ambient intelligent home care systems. In *Pervasive Health Conference and Workshops, 2006* (Innsbruck, Austria, 2006), pp. 1–10.

[19] Belatar, M., and Poirier, F. Text entry for mobile devices and users with severe motor impairments: Handiglyph, a primitive shapes based onscreen keyboard. In *Proceedings of the 10th International ACM SIGACCESS Conference on Computers and Accessibility* (New York, NY, USA, 2008), Assets '08, ACM, pp. 209–216.

[20] Bhuiyan, M., and Picking, R. Gesture-controlled user interfaces, what have we done and what's next? Tech. rep., Centre for Applied Internet Research (CAIR), Wrexham, UK, 2009.

[21] Biswas, P., and Langdon, P. Developing multimodal adaptation algorithm for mobility impaired users by evaluating their hand strength. *Int. J. Hum. Comput. Interaction 28*, 9 (2012), 576–596.

[22] Blackstock, M., Lea, R., and Krasic, C. Toward wide area interaction with ubiquitous computing environments. In *Proceedings of the First European conference on Smart Sensing and Context* (Berlin, Heidelberg, 2006), EuroSSC'06, Springer-Verlag, pp. 113–127.

[23] Bohn, J., Coroama, V., Langheinrich, M., Mattern, F., and Rohs, M. Social, economic, and ethical implications of ambient intelligence and ubiquitous computing. In *Ambient Intelligence*, W. Weber, J. Rabaey, and E. Aarts, Eds. Springer Berlin Heidelberg, 2005, pp. 5–29.

[24] Bruegger, P., and Hirsbrunner, B. Kinetic user interface: Interaction through motion for pervasive computing systems. In *Universal Access in Human-Computer Interaction. Intelligent and Ubiquitous Interaction Environments*, C. Stephanidis, Ed., vol. 5615 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2009, pp. 297–306.

[25] Buxton, W. There's more to interaction than meets the eye: Some issues in manual input. In *User Centered System Design: New Perspectives on Human-Computer Interaction*, D. A. Norman and S. W. Draper, Eds. Erlbaum, Hillsdale, NJ, 1986, pp. 319–337.

[26] Calvary, G., Coutaz, J., and Thevenin, D. Embedding plasticity in the development process of interactive systems. In *In 6th ERCIM Workshop "User Interfaces for All". Also in HUC (Handheld and Ubiquitous Computing) First workshop on Resource Sensitive Mobile HCI, Conference on Handheld and Ubiquitous Computing* (Florence, Italy, October 25-26 2000).

[27] Camurri, A., Hashimoto, S., Ricchetti, M., Ricci, A., Suzuki, K., Trocca, R., and Volpe, G. Eyesweb: Toward gesture and affect recognition in interactive dance and music systems. *Comput. Music J. 24*, 1 (Apr. 2000), 57–69.

[28] Camurri, A., Ricchetti, M., and Trocca, R. EyesWeb-toward gesture and affect recognition in dance/music interactive systems. In *Multimedia Computing and Systems, 1999. IEEE International Conference on* (Florence, Italy, Jul 1999), vol. 1, pp. 643–648 vol.1.

[29] Card, S. K., Mackinlay, J. D., and Robertson, G. G. A morphological analysis of the design space of input devices. *ACM Trans. Inf. Syst. 9* (April 1991), 99–122.

[30] Carlson, D., Altakrouri, B., and Schrader, A. Ambientweb: Bridging the web's cyber-physical gap. In *Internet of Things (IOT), 2012 3rd International Conference on the* (Wuxi, China, Oct 2012), pp. 1–8.

[31] Carlson, D., Altakrouri, B., and Schrader, A. An ad-hoc smart gateway platform for the web of things. In *Green Computing and Communications (GreenCom), 2013 IEEE and Internet of Things (iThings/CPSCom), IEEE International Conference on and IEEE Cyber, Physical and Social Computing* (Beijing, August 20-23 2013), pp. 619–625.

[32] Carlson, D., Altakrouri, B., and Schrader, A. Reinventing the share button for physical spaces. In *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2013 IEEE International Conference on* (San Diego, California, , USA, March 18-22 2013), pp. 318–320.

[33] Carlson, D., and Schrader, A. Dynamix: An open plug-and-play context framework for android. In *Proceedings of the 3rd International Conference on the Internet of Things (IoT2012)* (Wuxi, China, Oct 2012).

[34] Chen, M., AlRegib, G., and Juang, B.-H. 6dmg: a new 6d motion gesture database. In *Proceedings of the 3rd Multimedia Systems Conference* (New York, NY, USA, 2012), MMSys '12, ACM, pp. 83–88.

[35] Chi, D., Costa, M., Zhao, L., and Badler, N. The emote model for effort and shape. In *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques* (New York, NY, USA, 2000), SIGGRAPH '00, ACM Press/Addison-Wesley Publishing Co., pp. 173–182.

[36] Chi, D., Costa, M., Zhao, L., and Badler, N. The emote model for effort and shape. In *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques* (New York, NY, USA, 2000), SIGGRAPH '00, ACM Press/Addison-Wesley Publishing Co., pp. 173–182.

[37] Consortium, I.-M. ist-music: Context-aware self-adaptive platform for mobile applications, 2010.

[38] Cook, D., and Das, S. *Smart Environments: Technology, Protocols and Applications (Wiley Series on Parallel and Distributed Computing).* Wiley-Interscience, 2004.

[39] Corchado, J., Bajo, J., and Abraham, A. Gerami: Improving healthcare delivery in geriatric residences. *Intelligent Systems, IEEE 23*, 2 (2008), 19–25.

[40] da Silva, F. S. C., and Vasconcelos, W. W. Managing responsive environments with software agents. *Appl. Artif. Intell. 21*, 4-5 (Apr. 2007), 469–488.

[41] Dael, N., Mortillaro, M., and Scherer, K. The body action and posture coding system (bap): Development and reliability. *Journal of Nonverbal Behavior 36*, 2 (2012), 97–121.

[42] Daiber, F., Schöning, J., and Krüger, A. Towards a framework for whole body interaction with geospatial data. In *Whole Body Interaction*, D. England, Ed., Human-Computer Interaction Series. Springer London, 2011, pp. 197–207.

[43] Damjanovic, V. An ambient intelligent prototype for collaboration. *Encyclopedia of E-Collaboration, IGI Global* (October 2008), 29–35.

[44] Davies, N., and Gellersen, H.-W. Beyond prototypes: challenges in deploying ubiquitous systems. *Pervasive Computing, IEEE 1*, 1 (jan.-march 2002), 26 –35.

[45] Dey, A. K. Understanding and using context. *Personal Ubiquitous Comput. 5*, 1 (Jan. 2001), 4–7.

[46] Dourish, P. *Where the Action Is: The Foundations of Embodied Interaction (Bradford Books)*, new ed ed. The MIT Press, Aug. 2004.

[47] DUCATEL, K., BOGDANOWICZ, M., SCAPOLO, F., LEIJTEN, J., and BURGELMAN, J.-C. Scenarios for ambient intelligence in 2010. Tech. rep., IST Advisory Group Final Report, European Commission, EC, Brussels, 2001.

[48] Edwards, W. K., Newman, M. W., and Poole, E. S. The infrastructure problem in HCI. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2010), CHI '10, ACM, pp. 423–432.

[49] El-Saddik, A., Rahman, M., and Hossain, M. Authoring multimedia objects in collaborative ambient intelligent virtual environment. In *Haptic Audio Visual Environments and their Applications, 2005. IEEE International Workshop on* (Ottawa, Ontario, Canada, October 1-2 2005).

[50] Endres, C., Butz, A., and MacWilliams, A. A survey of software infrastructures and frameworks for ubiquitous computing. *Mob. Inf. Syst. 1*, 1 (Jan. 2005), 41–80.

[51] England, D. Whole body interactions: An introduction. In *Whole Body Interaction*, D. England, Ed. Springer London, 2011, ch. Whole Body Interactions: An Introduction, pp. 1–5.

[52] Explained, E. S. Population structure and ageing. Online, 31/10/2013 2013.

[53] Fagerberg, G., and Kung, A. Platforms for aal applications - europe support for aal systems. In *Smart Sensing and Context*, P. Lukowicz, K. Kunze, and G. Kortuem, Eds., vol. 6446 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2010, pp. 177–201.

[54] Fagerberg, P., Ståhl, A., and Höök, K. Designing gestures for affective input : an analysis of shape, effort and valence. In *Proceedings of the 2nd International Conference on Mobile and Ubiquitous Multimedia (MUM 2003), 10-12 December, 2003, Norrköping, Sweden* (2003), ACM, pp. 57–65.

[55] Fernandes, C & Bermudez, B. i. . A. p. Inventing the interactive glossary: an approach to documenting contemporary dance. *Notation: special issue of RTRSRCH Journal, Amsterdam Vol.2 No.2* (2010), 29–31.

[56] Fogtmann, M. H., Fritsch, J., and Kortbek, K. J. Kinesthetic interaction: revealing the bodily potential in interaction design. In *Proceedings of the 20th Australasian Conference on Computer-Human Interaction: Designing for Habitus and Habitat* (New York, NY, USA, 2008), OZCHI '08, ACM, pp. 89–96.

[57] Foley, J. D., Wallace, V. L., and Chan, P. The human factors of computer graphics interaction techniques. *IEEE Comput. Graph. Appl. 4*, 11 (Nov. 1984), 13–48.

[58] Freeman, W. T., and Weissman, C. D. Television control by hand gestures. In *International Workshop on Automatic Face and Gesture Recognition* (Zurich, Switzerland, June 26-28 1995), pp. 179–183.

[59] Froehlich, J., Wobbrock, J. O., and Kane, S. K. Barrier pointing: Using physical edges to assist target acquisition on mobile device touch screens. In *Proceedings of the 9th International ACM SIGACCESS Conference on Computers and Accessibility* (New York, NY, USA, 2007), Assets '07, ACM, pp. 19–26.

[60] Gárate, A., Herrasti, N., and López, A. Genio: an ambient intelligence application in home automation and entertainment environment. In *Proceedings of the 2005 joint conference on Smart objects and ambient intelligence: innovative context-aware services: usages and technologies* (New York, NY, USA, 2005), sOc-EUSAI '05, ACM, pp. 241–245.

[61] Gary Halajian, J. W. Gesture recognition based on scratch inputs. Tech. rep., Cornell University, Ithaca, NY, United States, April 26, 2009 2009. ECE 4760.

[62] Gerling, K., Livingston, I., Nacke, L., and Mandryk, R. Full-body motion-based game interaction for older adults. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2012), CHI '12, ACM, pp. 1873–1882.

[63] Gröschner, J. Motion-based interactions editor using labanotation. Bachelors thesis, Institute of Telematics at the University of Luebeck, Luebeck, Germany, November 2012.

[64] Gross, T. Ambient interfaces in a web-based theatre of work. In *Parallel, Distributed and Network-based Processing, 2002. Proceedings. 10th Euromicro Workshop on* (Canary Islands, Spain, Jan 1-11 2002), pp. 55–62.

[65] Gross, T. Ambient interfaces: Design challenges and recommendations. In *Proceedings of the 10th International Conference on Human-Computer Interaction - HCII 2003* (Crete, Greece, June 22-27 2003), pp. 68–72.

[66] Hachimura, K., and Nakamura, M. Method of generating coded description of human body motion from motion-captured data. In *10th IEEE International Workshop on Robot and Human Interactive Communication* (Bordeaux, Paris, France, Sep 18 -21 2001).

[67] Hamon, A., Palanque, P., Silva, J. L., Deleris, Y., and Barboni, E. Formal description of multi-touch interactions. In *Proceedings of the 5th ACM SIGCHI Symposium on Engineering*

Interactive Computing Systems (New York, NY, USA, 2013), EICS '13, ACM, pp. 207–216.

[68] Harrison, B. L., Fishkin, K. P., Gujar, A., Mochon, C., and Want, R. Squeeze me, hold me, tilt me! an exploration of manipulative user interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 1998), CHI '98, ACM Press/Addison-Wesley Publishing Co., pp. 17–24.

[69] Hashim, W. N. W., Noor, N. L. M., and Adnan, W. A. W. The design of aesthetic interaction: Towards a graceful interaction framework. In *Proceedings of the 2Nd International Conference on Interaction Sciences: Information Technology, Culture and Human* (New York, NY, USA, 2009), ICIS '09, ACM, pp. 69–75.

[70] Hatol, J. Movementxml: A representation of semantics of human movement based on labanotation. Master's thesis, SIMON FRASER UNIVERSITY, Burnaby, BC, Canada, 2006.

[71] Hinckley, K. Input technologies and techniques. In *The human-computer interaction handbook*, J. A. Jacko and A. Sears, Eds. L. Erlbaum Associates Inc., Hillsdale, NJ, USA, 2003, ch. Input technologies and techniques, pp. 151–168.

[72] Hinckley, K., Pierce, J., Sinclair, M., and Horvitz, E. Sensing techniques for mobile interaction. In *Proceedings of the 13th Annual ACM Symposium on User Interface Software and Technology* (New York, NY, USA, 2000), UIST '00, ACM, pp. 91–100.

[73] Hoare, T., and Milner, R., Eds. *Grand challenges in computing research*. The British Computer Society, Swindon, 2004.

[74] Hoste, L. Software engineering abstractions for the multi-touch revolution. In *Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering - Volume 2* (New York, NY, USA, 2010), ICSE '10, ACM, pp. 509–510.

[75] Hummels, C., Overbeeke, K. C., and Klooster, S. Move to get moved: a search for methods, tools and knowledge to design for expressive and rich movement-based interaction. *Personal Ubiquitous Comput. 11*, 8 (Dec. 2007), 677–690.

[76] Hurst, A., Hudson, S. E., Mankoff, J., and Trewin, S. Automatically detecting pointing performance. In *Proceedings of the 13th International Conference on Intelligent User Interfaces* (New York, NY, USA, 2008), IUI '08, ACM, pp. 11–19.

[77] Hutchinson, A. *Labanotation The System of Analyzing and Recording Movement*, 4th ed. Routledge, NewYork and London, 2005.

[78] Hwang, F., Keates, S., Langdon, P., and Clarkson, P. J. Multiple haptic targets for motion-impaired computer users. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2003), CHI '03, ACM, pp. 41–48.

[79] Iacolina, S., Lai, A., Soro, A., and Scateni, R. Natural interaction and computer graphics applications. In *Eurographics Italian Chapter Conference* (Genova, Italy, 2010), E. Puppo, A. Brogni, and L. D. Floriani, Eds., Eurographics Association, pp. 141–146.

[80] Ishii, H., Wisneski, C., Brave, S., Dahley, A., Gorbet, M., Ullmer, B., and Yarin, P. ambientroom: integrating ambient media with architectural space. In *CHI 98 Cconference Summary on Human Factors in Computing Systems* (New York, NY, USA, 1998), CHI '98, ACM, pp. 173–174.

[81] James, J., Ingalls, T., Qian, G., Olsen, L., Whiteley, D., Wong, S., and Rikakis, T. Movement-based interactive dance performance. In *Proceedings of the 14th Annual ACM International Conference on Multimedia* (New York, NY, USA, 2006), MULTIMEDIA '06, ACM, pp. 470–480.

[82] Jochems, N., Vetter, S., and Schlick, C. A comparative study of information input devices for aging computer users. *Behaviour & Information Technology 32*, 9 (2012), 1–18.

[83] Juchno, P., Marcu, M., and Vasileva, K. *Europe in figures - Eurostat yearbook 2010.* Eurostat, Statistical office of the European Union, Luxembourg, 2010, ch. Population, pp. 149 – 203.

[84] Kahol, K., Tripathi, P., and Panchanathan, S. Documenting motion sequences with a personalized annotation system. *IEEE MultiMedia 13*, 1 (2006), 37–45.

[85] Kallio, S., Kela, J., Mäntyjärvi, J., and Plomp, J. Visualization of hand gestures for pervasive computing environments. In *Proceedings of the working conference on Advanced visual interfaces* (New York, NY, USA, 2006), AVI '06, ACM, pp. 480–483.

[86] Kane, S. K., Jayant, C., Wobbrock, J. O., and Ladner, R. E. Freedom to roam: a study of mobile device adoption and accessibility for people with visual and motor disabilities. In *Proceedings of the 11th international ACM SIGACCESS conference on Computers and accessibility* (New York, NY, USA, 2009), Assets '09, ACM, pp. 115–122.

[87] Karam, M., Hare, J., Lewis, P., and Schraefel, M. Ambient gestures. Tech. rep., Intelligence, Agents, Multimedia Group, University of Southampton., Southampton, United Kingdom, 2006.

[88] Karam, M., and m. c. schraefel. A taxonomy of gestures in human computer interactions. Technical report, University of Southampton, Southampton, United Kingdom, 2005.

[89] Kasahara, S., Niiyama, R., Heun, V., and Ishii, H. extouch: Spatially-aware embodied manipulation of actuated objects mediated by augmented reality. In *Proceedings of the 7th International Conference on Tangible, Embedded and Embodied Interaction* (New York, NY, USA, 2013), TEI '13, ACM, pp. 223–228.

[90] Kauw-A-Tjoe, R., Thalen, J., Marin-Perianu, M., and Havinga, P. Sensorshoe: Mobile gait analysis for parkinson's disease patients. In *UbiComp 2007 Workshop Proceedings* (September 2007), University of Innsbruck, pp. 187–191.

[91] Kela, J., Korpipää, P., Mäntyjärvi, J., Kallio, S., Savino, G., Jozzo, L., and Marca, D. Accelerometer-based gesture control for a design environment. *Personal Ubiquitous Comput. 10*, 5 (July 2006), 285–299.

[92] KEMPPAINEN, E., ABASCAL, J., ALLEN, B., DELAITRE, S., GIOVANNINI, C., and SOEDE, M. *Towards an inclusive future Impact and wider potential of information and communication technologies.* COST, Brussels, 2007, ch. Ethical and legislative issues with regard to Ambient Intelligence, pp. 188 – 205.

[93] Khandkar, S. H., and Maurer, F. A language to define multi-touch interactions. In *ACM International Conference on Interactive Tabletops and Surfaces* (New York, NY, USA, 2010), ITS '10, ACM, pp. 269–270.

[94] Kim, J.-W., and Nam, T.-J. Eventhurdle: supporting designers' exploratory interaction prototyping with gesture-based sensors. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2013), CHI '13, ACM, pp. 267–276.

[95] Kin, K., Hartmann, B., DeRose, T., and Agrawala, M. Proton++: a customizable declarative multitouch framework. In *Proceedings of the 25th annual ACM symposium on User interface software and technology* (New York, NY, USA, 2012), UIST '12, ACM, pp. 477–486.

[96] Kjeldsen, R., and Kender, J. Toward the use of gesture in traditional user interfaces. In *Proceedings of the 2nd International Conference on Automatic Face and Gesture Recognition (FG '96)* (Washington, DC, USA, 1996), FG '96, IEEE Computer Society, pp. 151–.

[97] Kleinberger, T., Becker, M., Ras, E., Holzinger, A., and Müller, P. Ambient intelligence in assisted living: Enable elderly people to handle future interfaces. In *Universal Access in Human-Computer Interaction. Ambient Interaction*, C. Stephanidis, Ed., vol. 4555 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2007, pp. 103–112.

[98] Kojima, K., Hachimura, K., and Nakamura, M. Labaneditor: Graphical editor for dance notation. In *In Proceedings of the 11th IEEE International Workshop on Robot and Human Interactive Communication.* (Berlin, Germany, Sep 25 - 27 2002), pp. 59 – 64.

[99] Kung, A., and Furfari, F. Platforms for aal applications - challenges for the building of aal platforms. In *Smart Sensing and Context*, P. Lukowicz, K. Kunze, and G. Kortuem, Eds., vol. 6446 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2010, pp. 177–201.

[100] Larssen, A., Loke, L., Robertson, T., and Edwards, J. Understanding movement as input for interaction a study of two eyetoy[a] games. In *OZCHI 2004, ESA and CHISIG* (Wollongong, Australia, Nov 22 - 24 2004).

[101] Lee, S., Buxton, W., and Smith, K. C. A multi-touch three dimensional touch-sensitive tablet. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 1985), CHI '85, ACM, pp. 21–25.

[102] Linjama, J., Korpipää, P., Kela, J., and Rantakokko, T. Actioncube: a tangible mobile gesture interaction tutorial. In *Proceedings of the 2nd International Conference on Tangible and Embedded Interaction* (New York, NY, USA, 2008), TEI '08, ACM, pp. 169–172.

[103] Lissermann, R., Huber, J., Hadjakos, A., and Mühlhäuser, M. Earput: Augmenting behind-the-ear devices for ear-based interaction. In *CHI '13 Extended Abstracts on Human Factors in Computing Systems* (New York, NY, USA, 2013), CHI EA '13, ACM, pp. 1323–1328.

[104] Loke, L., Larssen, A. T., and Robertson, T. Labanotation for design of movement-based interaction. In *Proceedings of the 2nd Australasian Conference on Interactive Entertainment* (Sydney, Australia, 2005), IE 2005, Creativity & Cognition Studios Press, pp. 113–120.

[105] Loke, L., Larssen, A. T., Robertson, T., and Edwards, J. Understanding movement for interaction design: frameworks and approaches. *Personal and Ubiquitous Computing 11*, 8 (2006), 691–701.

[106] Lü, H., and Li, Y. Gesture coder: a tool for programming multi-touch gestures by demonstration. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2012), CHI '12, ACM, pp. 2875–2884.

[107] Lü, H., and Li, Y. Gesture studio: authoring multi-touch interactions through demonstration and declaration. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2013), CHI '13, ACM, pp. 257–266.

[108] Luczak, H., Schlick, C., Jochems, N., Vetter, S., and Kausch, B. *Touch Screens for the Elderly: Some Models and Methods, Prototypical Development and Experimental Evaluation of Human-Computer Interaction Concepts for the Elderly.* D. Haftor and A. Mirijamdotter, 2011, pp. 116–135.

[109] Lyytinen, K., and Yoo, Y. Issues and challenges in ubiquitous computing. *Commun. ACM 45*, 12 (Dec. 2002), 62–65.

[110] Mackinlay, J., Card, S. K., and Robertson, G. G. A semantic analysis of the design space of input devices. *Hum.-Comput. Interact. 5* (June 1990), 145–190.

[111] Mahmud, M., and Kurniawan, H. Involving psychometric tests for input device evaluation with older people. In *Proceedings of the 17th Australia conference on Computer-Human Interaction: Citizens Online: Considerations for Today and the Future* (Narrabundah, Australia, Australia, 2005), OZCHI '05, Computer-Human Interaction Special Interest Group (CHISIG) of Australia, pp. 1–10.

[112] Malizia, A., and Bellucci, A. The artificiality of natural user interfaces. *Commun. ACM 55*, 3 (Mar. 2012), 36–38.

[113] Mardanbegi, D., Hansen, D. W., and Pederson, T. Eye-based head gestures. In *Proceedings of the Symposium on Eye Tracking Research and Applications* (New York, NY, USA, 2012), ETRA '12, ACM, pp. 139–146.

[114] Marinos, D., Wöldecke, B., and Geiger, C. Prototyping natural interactions in virtual studio environments by demonstration: combining spatial mapping with gesture following. In *Proceedings of the Virtual Reality International Conference: Laval Virtual* (New York, NY, USA, 2013), VRIC '13, ACM, pp. 2:1–2:8.

[115] McArthur, V., Castellucci, S. J., and MacKenzie, I. S. An empirical comparison of "wiimote" gun attachments for pointing tasks. In *Proceedings of the 1st ACM SIGCHI symposium on Engineering interactive computing systems* (New York, NY, USA, 2009), EICS '09, ACM, pp. 203–208.

[116] McKinley, P., Sadjadi, S., Kasten, E., and Cheng, B. H. C. Composing adaptive software. *Computer 37*, 7 (July 2004), 56–64.

[117] McMahan, R. P., Alon, A. J. D., Lazem, S., Beaton, R. J., Machaj, D., Schaefer, M., Silva, M. G., Leal, A., Hagan, R., and Bowman, D. A. Evaluating natural interaction techniques in video games. In *Proceedings of the 2010 IEEE Symposium on 3D User Interfaces* (Washington, DC, USA, 2010), 3DUI '10, IEEE Computer Society, pp. 11–14.

[118] Moen, J. Kinaesthetic movement interaction: Designing for the pleasure of motion. Ph.d. dissertation, KTH, Numerical Analysis and Computer Science, Sweden, 2006.

[119] Moen, J. From hand-held to body-worn: Embodied experiences of the design and use of a wearable movement-based interaction concept. In *Proceedings of the 1st International Conference on Tangible and Embedded Interaction* (New York, NY, USA, 2007), TEI '07, ACM, pp. 251–258.

[120] Mulder, A. Hand gestures for hci. *Hand Centered Studies of Human Movement Project* (1996), 1–21.

[121] Mysliwiec, T. Fingermouse: A freehand computer pointing interface. In *in Proc. of Intl Conf. on Automatic Face and Gesture Recognition* (1994), pp. 372–377.

[122] Navarre, D., Palanque, P., Ladry, J.-F., and Barboni, E. ICOs: A model-based user interface description technique dedicated to interactive systems addressing usability, reliability and scalability. *ACM Trans. Comput.-Hum. Interact. 16*, 4 (Nov. 2009), 18:1–18:56.

[123] Nielsen, J. *Usability Engineering.* Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1993.

[124] Niemelä, E., and Latvakoski, J. Survey of requirements and solutions for ubiquitous software. In *Proceedings of the 3rd International Conference on Mobile and Ubiquitous Multimedia* (New York, NY, USA, 2004), MUM '04, ACM, pp. 71–78.

[125] Niezen, G. *Ontologies for interaction: enabling serendipitous interoperability in smart environments.* Phd thesis, Technische Universiteit Eindhoven, Department of Industrial Design, Eindhoven, The Netherlands, June 2012.

[126] Norman, D. A. Natural user interfaces are not natural. *interactions 17*, 3 (May 2010), 6–10.

[127] Nugent, C. D., Galway, L., Chen, L., Donnelly, M. P., McClean, S. I., Zhang, S., Scotney, B. W., and Parr, G. Managing sensor data in ambient assisted living. *JCSE 5*, 3 (2011), 237–245.

[128] Offermans, M. Software Architecture - Apache ACE. Online (https://cwiki.apache.org/confluence/display/ACE/Software+Architecture), 2009.

[129] Organization, W. H. *International classification of functioning, disability and health : ICF.* Geneva : World Health Organization, 2001, 2001.

[130] Organization, W. H. *How to use the ICF A Practical Manual for using the International Classification of Functioning, Disability and Health (ICF). Exposure draft for comment.* Geneva : World Health Organization, 2013, 2013.

[131] Papadopoulou, E., Gallacher, S., Taylor, N. K., and Williams, M. H. A personal smart space approach to realising ambient ecologies. *Pervasive Mob. Comput. 8*, 4 (Aug. 2012), 485–499.

[132] Paterno, F. *User Interface Design Adaptation.* The Interaction Design Foundation, Aarhus, Denmark, 2013, ch. User Interface Design Adaptation.

[133] Pattison, M., and Stedmon, A. W. Inclusive design and human factors: Designing mobile phones for older users. *PsychNology Journal* (2006), 267–284.

[134] Piper, A. M., Campbell, R., and Hollan, J. D. Exploring the accessibility and appeal of surface computing for older adult health care support. In *Proceedings of the 28th International Conference on Human Factors in Computing Systems* (New York, NY, USA, 2010), CHI '10, ACM, pp. 907–916.

[135] Press, M. Microsoft facility helps you make yourself at home in the future. Online (http://www.microsoft.com/en-us/news/features/2011/aug11/08-08microsofthome.aspx), August 2011.

[136] Priyantha, N. B., Chakraborty, A., and Balakrishnan, H. The cricket location-support system. In *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking* (New York, NY, USA, 2000), MobiCom '00, ACM, pp. 32–43.

[137] Pruvost, G., Heinroth, T., Bellik, Y., and Minker, W. *User Interaction Adaptation within Ambient Environments*, next generation intelligent environments: ambient adaptive systems ed. Springer, Boston (USA), 2011, ch. 5, pp. 153–194.

[138] Quek, F. The catchment feature model: a device for multimodal fusion and a bridge between signal and sense. *EURASIP J. Appl. Signal Process. 2004* (Jan. 2004), 1619–1636.

[139] Rahman, M., Gustafson, S., Irani, P., and Subramanian, S. Tilt techniques: investigating the dexterity of wrist-based input. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2009), CHI '09, ACM, pp. 1943–1952.

[140] Rekimoto, J. Tilting operations for small screen interfaces. In *Proceedings of the 9th Annual ACM Symposium on User Interface Software and Technology* (New York, NY, USA, 1996), UIST '96, ACM, pp. 167–168.

[141] Rico, J., and Brewster, S. Usable gestures for mobile interfaces: evaluating social acceptability. New York, NY, USA, 2010.

[142] Román, M., Hess, C., Cerqueira, R., Ranganathan, A., Campbell, R. H., and Nahrstedt, K. Gaia: a middleware platform for active spaces. *SIGMOBILE Mob. Comput. Commun. Rev. 6*, 4 (Oct. 2002), 65–67.

[143] Rousseau, C., Bellik, Y., Vernier, F., and Bazalgette, D. A framework for the intelligent multimodal presentation of information. *Signal Process. 86*, 12 (Dec. 2006), 3696–3713.

[144] Rouvroy, A. Privacy, data protection, and the unprecedented challenges of ambient inteligence. *Studies in Ethics, Law, and Technology, Berkeley Electronic Press* (2008).

[145] Ruiz, J., and Li, Y. Doubleflip: a motion gesture delimiter for mobile interaction. In *Proceedings of the 2011 annual conference on Human factors in computing systems* (New York, NY, USA, 2011), CHI '11, ACM, ACM, p. 2717–2720.

[146] Ruiz, J., Li, Y., and Lank, E. User-defined motion gestures for mobile interaction. In *Proceedings of the 2011 annual conference on Human factors in computing systems* (New York, NY, USA, 2011), CHI '11, ACM, pp. 197–206.

[147] Sadri, F. Ambient intelligence: A survey. *ACM Comput. Surv. 43*, 4 (Oct. 2011), 36:1–36:66.

[148] Saha, D., and Mukherjee, A. Pervasive computing: a paradigm for the 21st century. *Computer 36*, 3 (Mar 2003), 25–31.

[149] Schiphorst, T., Lovell, R., and Jaffe, N. Using a gestural interface toolkit for tactile input to a dynamic virtual space. In *CHI'02 Extended Abstracts on Human Factors in Computing Systems* (New York, NY, USA, 2002), CHI EA '02, ACM, pp. 754–755.

[150] Scholliers, C., Hoste, L., Signer, B., and De Meuter, W. Midas: a declarative multi-touch interaction framework. In *Proceedings of the fifth International Conference on Tangible, Embedded, and Embodied Interaction* (New York, NY, USA, 2011), TEI '11, ACM, pp. 49–56.

[151] Schöning, J., Daiber, F., Krüger, A., and Rohs, M. Using hands and feet to navigate and manipulate spatial data. In *CHI '09 Extended Abstracts on Human Factors in Computing Systems* (New York, NY, USA, 2009), CHI EA '09, ACM, pp. 4663–4668.

[152] Schrader, A., Carlson, D. V., and Busch, D. Modular framework support for context-aware mobile cinema. *Personal Ubiquitous Comput. 12*, 4 (Feb. 2008), 299–306.

[153] Schrader, A., Rothenpieler, P., and Träder, S. F. . J.-M. Ambient socio-technical support for assisted autonomous living. *ERCIM News, Special Issue on Ambient Assisted Living*, 87 (October 2011).

[154] Scoditti, A., Blanch, R., and Coutaz, J. A novel taxonomy for gestural interaction techniques based on accelerometers. In *the 15th International Conference on Intelligent User Interfaces (IUI '11)* (New York, NY, USA, 2011), ACM, pp. 63–72.

[155] Sears, A., Lin, M., Jacko, J., and Xiao, Y. When Computers Fade...Pervasive Computing and Situationally-Induced Impairments and Disabilities. In *Human-Computer Interaction: Theory and Practice (Part II)*, C. Stephanidis, Ed. Lawrence Erlbaum Associates, Hillsdale, 2003, pp. 1298–1302.

[156] Sears, A., and Young, M. Physical disabilities and computing technologies: An analysis of impairments. In *The Human-computer Interaction Handbook*, J. A. Jacko and A. Sears, Eds. L. Erlbaum Associates Inc., Hillsdale, NJ, USA, 2003, ch. Physical Disabilities and Computing Technologies: An Analysis of Impairments, pp. 482–503.

[157] Si, H., Kawahara, Y., Morikawa, H., and Aoyama, T. A stochastic approach for creating context-aware services based on context histories in smart home. In *Proceeding of the 3rd International Conference on Pervasive Computing, Exploiting Context Histories in Smart Environments* (Munich, Germany, May 2005), pp. 37–41.

[158] Smigel, L., Goldstein, M., Aldrich, E., Owen, N., and Drazin, B. *Documenting Dance: A Practical Guide.* Dance Heritage Coalition, Incorporated, 2006.

[159] Society, E. C. I., and Media. Overview of the european strategy in ict for ageing. Online, October 2010.

[160] Sparacino, F. Natural interaction in intelligent spaces: Designing for architecture and entertainment. *Multimedia Tools Appl. 38*, 3 (2008), 307–335.

[161] Stivers, T., and Sidnell, J. Introduction: Multimodal interaction. *Semiotica Volume 2005, Issue 15* (2005), 1–227.

[162] Sutterer, M., Coutand, O., Droegehorn, O., David, K., and van der Sluijs, K. Managing and delivering context-dependent user preferences in ubiquitous computing environments. In *International Symposium on Applications and the Internet - Workshops (SAINT 2007 Workshops)* (Hiroshima, Japan., January 15-19 2007), pp. 4–4.

[163] Trewin, S., Keates, S., and Moffatt, K. Developing steady clicks:: A method of cursor assistance for people with motor impairments. In *Proceedings of the 8th International ACM SIGACCESS Conference on Computers and Accessibility* (New York, NY, USA, 2006), Assets '06, ACM, pp. 26–33.

[164] Triesch, J., and Malsburg, C. Robotic gesture recognition. In *Gesture and Sign Language in Human-Computer Interaction*, I. Wachsmuth and M. Fröhlich, Eds., vol. 1371 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 1998, pp. 233–244.

[165] Ullmer, B., and Ishii, H. Emerging frameworks for tangible user interfaces. *IBM Syst. J. 39*, 3-4 (July 2000), 915–931.

[166] van der Vlist, B., Niezen, G., Hu, J., and Feijs, L. M. G. Interaction primitives: Describing interaction capabilities of smart objects in ubiquitous computing environments. In *AFRICON, 2011* (2011), pp. 1–6.

[167] Vatavu, R.-D. Interfaces that should feel right: Natural interaction with multimedia information. In *Recent Advances in Multimedia Signal Processing and Communications*, M. Grgic, K. Delac, and M. Ghanbari, Eds., vol. 231 of *Studies in Computational Intelligence*. Springer Berlin Heidelberg, 2009, pp. 145–170.

[168] Vatavu, R.-D. Understanding challenges in designing interactions for the age of ambient media. In *The 3rd Workshop on Semantic Ambient Media Experience (SAME) at AmI 2010* (Malaga, Nov. 2010 2010), pp. 8–13.

[169] Velastin, S., Khoudour, L., Lo, B. P. L., Sun, J., and Vicencio-Silva, M. A. PRISMATICA: a multi-sensor surveillance system for public transport networks. In *12th IEE International Conference on Road Transport Information and Control, 2004. RTIC 2004* (London, UK, April 20-22 2004), pp. 19–25.

[170] Vogel, D., and Balakrishnan, R. Interactive public ambient displays: transitioning from implicit to explicit, public to personal, interaction with multiple users. In *Proceedings of the 17th annual ACM symposium on User interface software and technology* (New York, NY, USA, 2004), UIST '04, ACM, pp. 137–146.

[171] Wacharamanotham, C., Hurtmanns, J., Mertens, A., Kronenbuerger, M., Schlick, C., and Borchers, J. Evaluating swabbing: A touchscreen input method for elderly users with tremor. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2011), CHI '11, ACM, pp. 623–626.

[172] Wachs, J. P., Kölsch, M., Stern, H., and Edan, Y. Vision-based hand-gesture applications. *Commun. ACM 54* (February 2011), 60–71.

[173] Wallbott, H. G. Bodily expression of emotion. *European journal of social psychology 28*, 6 (1998), 879–896.

[174] Wang, A. I., and Ahmad, Q. K. Camf - context-aware machine learning framework for android. In *Iasted International Conference on Software Engineering and Applications (SEA 2010)* (CA, USA, 2010, November 8-10 2010), M. D. Rey, Ed.

[175] Want, R., and Pering, T. System challenges for ubiquitous pervasive computing. In *27th International Conference on Software Engineering (ICSE 2005)* (St. Louis, MO, USA, May 15 - 21 2005), pp. 9 – 14.

[176] WebAIM. Motor disabilities:assistive technologies. http://webaim.org/articles/motor/assistive, 2011. Accessed November 01, 2011.

[177] Weber, W., Rabbaey, J., and Aarts, E., Eds. *Ambient Intelligence.* Springer Verlag, Berlin, 2005.

[178] Weiser, M. Some computer science issues in ubiquitous computing. *Commun. ACM 36*, 7 (July 1993), 75–84.

[179] Weiser, M. The computer for the 21st century. *Scientific American, 265, No. 3* (September199l), 94–104.

[180] Williams, M. H., Taylor, N. K., Roussaki, I., Farshchian, B., and Doolin, K. Developing a pervasive system for a mobile environments. In *eChallenges 2006 - Exploiting the Knowledge Economy* (2006), IOS Press, pp. 1695 – 1702.

[181] Wilson, F. *The Hand: How Its Use Shapes the Brain, Language, and Human Culture.* Vintage Series. Vintage Books, 1999.

[182] WOBBROCK, J. O. The future of mobile device research in "hci". In *CHI 2006, "What is the Next Generation of Human-Computer Interaction?"* (Montreal, Quebec, Canada., April 22-27, 2006 2006).

[183] WOBBROCK, J. O., MORRIS, M. R., AND WILSON, A. D. User-defined gestures for surface computing. In *Proceedings of the 27th international conference on Human factors in computing systems* (New York, NY, USA, 2009), CHI '09, ACM, pp. 1083–1092.

[184] WOBBROCK, J. O., MYERS, B. A., AND KEMBEL, J. A. Edgewrite: A stylus-based text entry method designed for high accuracy and stability of motion. In *Proceedings of the 16th Annual ACM Symposium on User Interface Software and Technology* (New York, NY, USA, 2003), UIST '03, ACM, pp. 61–70.

[185] WOO, W., KIM, N., WONG, K., AND TADENUMA, M. Sketch on dynamic gesture tracking and analysis exploiting vision-based 3d interface. In *Photonics West 2001-Electronic Imaging* (2000), International Society for Optics and Photonics, pp. 656–666.

[186] WRIGHT, D., GUTWIRTH, S., FRIEDEWALD, M., VILDJIOUNAITE, E., AND PUNIE, Y., Eds. *Safeguards in a World of Ambient Intelligence.* The International Library of Ethics, Law and Technology, Springer, 2008.

[187] YEE, C. S. M. *Advanced and natural interaction system for motion-impaired users.* PhD thesis, Departament de Ciencies Matematiques i Informatica,Universitat de les Illes Balears , Spain, 2009.

[188] ZHOU, Y. MusicWalk: Gait training using mobile devices for parkinsons patients. Design of advanced user interfaces final project report, National University of Singapore, NUS, Singapore, 2009.