Designing expressive interaction techniques for novices inspired by expert activities : the case of musical practice

Emilien Ghomi

UNIVERSITÉ PARIS-SUD

ÉCOLE DOCTORALE : Informatique
Laboratoire de Recherche en Informatique
Équipe in-situ

DISCIPLINE : Informatique (Interaction Homme-Machine)

THÈSE DE DOCTORAT

Soutenue le 17/12/2012
par

ÉMILIEN GHOMI

DESIGNING EXPRESSIVE INTERACTION TECHNIQUES FOR NOVICES INSPIRED BY EXPERT ACTIVITIES: THE CASE OF MUSICAL PRACTICE

Co-directeurs de thèse :	Michel Beaudouin-Lafon	Prof. Univ. Paris-Sud, LRI-InSitu
	Stéphane Huot	MdC Univ. Paris-Sud, LRI-InSitu

Composition du jury :

Président du jury :	Christian Jacquemin	Prof. Univ. Paris-Sud, LIMSI-AMI
Rapporteurs :	Carlos Agon	Prof. Univ. Pierre et Marie Curie, IRCAM
	Marcelo Wanderley	Associate Prof. McGill Univ., Schulich Sch. of Music-CIRMMT
Examinateur :	Yannick Prié	Prof. Univ. de Nantes, LINA-KOD

Designing expressive interaction techniques for novices inspired by expert activities: The case of musical practice

SUPERVISORS: Michel Beaudouin-Lafon, Stéphane Huot

Paris, France, le 17 décembre 2012.

— À Beisha, l'ambre // À toutes mes sources d'inspiration —

REMERCIEMENTS

Je tiens tout d'abord à remercier mes directeurs de thèse, Michel et Stéphane, pour m'avoir permis de travailler sur cette thématique, et d'en explorer la diversité de manière personnelle. Merci pour vos conseils, vos critiques et votre exigence qui m'ont maintenu dans une bonne direction.

Merci à ceux avec qui j'ai collaboré : Guillaume, Hugues, les Olivier, Vincent et Wendy. La diversité de ce travail dépend grandement de la richesse de vos points de vue, de votre écoute et de votre disponibilité.

Merci à toutes les belles personnalités, devenues des amis, que j'ai pu rencontrer au labo et avec qui nous avons animé d'interminables discussions autour de la recherche et de tout le reste : Clément, Ilaria, Jérémie, Julie, Mathieu et Tony.

Merci à mon parrain et à ma marraine de recherche – Aurélien et Anne – qui m'ont grandement éveillé au domaine et ont débroussaillé, alimenté ou interdit les voies dans lesquelles je voulais m'engager.

Merci à ma famille, aux amis et au frère qui ont partagé mon toit pendant cette période et qui m'ont supporté, dans tous les sens du terme : Gaëtan, Ju et Emilie, Manue et Maud.

Merci aux amis qui m'ont suivi de près ou de loin dans ce travail et ont été des repères dont je n'aurais pu me passer : Cédric, Dun, Elise, Fanny, Florent, FX et Laura, Léa, Lise et Normy.

Merci à mes à-côtés, qui m'ont maintenu attentif à l'extérieur et ont toujours été les moteurs de ma motivation pour créer des ponts entre les disciplines. Sur les planches du théâtre. Sur la scène musicale. Dans l'événementiel en Suisse ou en France. Dans l'entreprenariat. Finalement tout est lié !

Abstract

As interactive systems are now used to perform a variety of complex tasks, users need systems that are at the same time *expressive*, *efficient* and *usable*. Although simple interactive systems can be easily *usable*, interaction designers often consider that only expert practitioners can benefit from the *expressiveness* of more complex systems.

Our approach, inspired by studies in phenomenology and psychology, underscores that non-experts have sizeable knowledge and advanced skills related to various expert activities having a social dimension –such as artistic activities–, which they gain implicitly through their engagement as perceivers. For example, we identify various music-related skills mastered by non-musicians, which they gain when listening to music or attending performances.

We have two main arguments. First, interaction designers can reuse such implicit knowledge and skills to design interaction techniques that are both *expressive* and *usable* by novice users.

Second, as expert artifacts and expert learning methods have evolved over time and have shown efficient to overcome the complexity of expert activities, they can be used as a source of inspiration to make *expressive* systems more easily *usable* by novice users.

We provide a design framework for studying the *usability* and *expressiveness* of interaction techniques as two new aspects of the user experience, and explore this framework with three projects.

In the first project we study the use of rhythmic patterns as an input method, and show that novice users are able to reproduce and memorize large vocabularies of patterns. This is made possible by the natural abilities of non-musicians to perceive, reproduce and make sense of rhythmic structures. We define a method to create *expressive* vocabularies of patterns, and show that novice users are able to efficiently use them as command triggers.

In the second project, we study the design and learning of chording gestures on multitouch screens. We introduce design guidelines to create *expressive* chord vocabularies taking the mechanical constraints and the degrees of freedom of the human hand into account. We evaluate the *usability* of such gestures in an experiment and we present an adapted learning method inspired by the teaching of chords in music. We show that novice users are able to reproduce and memorize our vocabularies of chording gestures, while our learning method can improve long-term memorization.

The final project focuses on music software used for live performances and proposes a framework for designing "instrumental" software allowing expert musical playing and having its elementary functionalities accessible to novices, as it is the case with acoustic instruments (for example, one can easily play a few chords on a piano without practice). We define a design framework inspired by a functional decomposition of acoustic instruments and present an adapted software architecture, both aiming to ease the design of such software and to make it match with instrument-making.

These projects show that, in these cases: (i) the implicit knowledge novices have about some expert activities can be reused for interaction; (ii) expert learning methods can inspire ways to make *expressive* systems more *usable* novices; (iii) taking expert artifacts as a source of inspiration can help creating *usable* and *expressive* interactive systems.

In this dissertation, we propose the study of *usability* as an alternative to the focus on immediacy that characterizes current commercial interactive systems. We also propose methods to benefit from the richness of expert activities and from the implicit knowledge of non-experts to design interactive systems that are at the same time *expressive* and *usable* by novice users.

Résumé

Les systèmes interactifs étant utilisés pour réaliser des tâches toujours plus complexes et variées, les utilisateurs ont besoin de systèmes qui soient à la fois *expressifs*, *efficaces* et *utilisables*. Si des systèmes simples peuvent être instantanément *utilisables*, l'*expressivité* accessible avec des systèmes complexes est souvent considérée comme réservée aux experts. Cependant, notre approche, inspirée par la recherche en phénoménologie et en psychologie, souligne que certaines activités expertes ayant une portée sociale, comme les activités artistiques, permettent aussi aux non-experts d'acquérir des compétences et une connaissance considérables de façon implicite. Dans ce manuscrit, nous évoquerons notamment la connaissance et les compétences avancées développées par les non-musiciens lors de l'écoute de la musique et de l'observation du jeu instrumental.

Nous défendons deux idées. Premièrement, les concepteurs de systèmes interactifs peuvent profiter de ces compétences et de cette connaissance implicites pour créer des systèmes *expressifs* qui soient *utilisables*.

Deuxièmement, les méthodes d'apprentissage expertes et les outils experts, qui ont été perfectionnés à travers le temps et ont fait leurs preuves dans des situations complexes, peuvent servir de sources d'inspiration pour améliorer l'*utilisabilité* des systèmes complexes pour les utilisateurs novices.

Nous proposons un cadre de conception pour étudier l'*utilisabilité* et l'*expressivité* des techniques d'interaction, comme deux nouvelles mesures de la qualité de l'interaction, et présentons les trois projets de cette thèse.

Dans le premier, nous étudions l'utilisation de motifs rythmiques pour l'interaction, et nous montrons que des utilisateurs novices sont capables de reproduire et de mémoriser efficacement de grands vocabulaires de motifs rythmiques. Une telle interaction tire parti des capacités naturelles des non-musiciens pour percevoir et reproduire des structures rythmiques. Nous définissons des règles pour créer des motifs rythmiques adaptés à l'interaction, et montrons qu'ils peuvent être utilisés efficacement pour déclencher des commandes.

Dans le deuxième projet, nous étudions la conception et l'apprentissage de postures multi-doigt sur des écrans multi-tactiles. Nous prenons en compte les contraintes mécaniques et les degrés de liberté de la main pour créer des vocabulaires expressifs de postures multi-doigt, dont nous évaluons l'*utilisabilité* lors d'une expérimentation. Nous présentons une méthode d'apprentissage adaptée aux postures les plus complexes, inspirée par l'apprentissage des accords en musique, et nous montrons qu'elle peut améliorer la compréhension et la mémorisation.

Dans le dernier projet, nous nous intéressons aux applications de création musicale en temps réel, et tentons de les faire profiter des qualités instrumentales des instruments acoustiques. Nous voulons créer des applications qui permettent un jeu virtuose et *expressif*, et dont les fonctionnalités élémentaires sont accessibles aux novices (comme on peut jouer quelques accords au piano sans apprentissage). Nous proposons un cadre de conception et une architecture logicielle qui aident à considérer la conception d'applications musicales comme une lutherie à part entière.

Avec ces projets, nous montrons que, dans ces cas : (i) la connaissance et les compétences implicites des non-experts peuvent être réutilisées en interaction ; (ii) les méthodes d'apprentissage expertes peuvent permettre de rendre les systèmes *expressifs* plus *utilisables* ; (iii) s'inspirer des outils experts peut aider à concevoir des systèmes interactifs *expressifs* et *utilisables*. Nous proposons l'étude de l'*utilisabilité* comme une alternative à l'immédiateté prônée par les entreprises d'informatique, et nous présentons des méthodes pour tirer parti de la richesse des activités expertes et de la connaissance implicite des non-experts pour créer des systèmes interactifs *expressifs* et *utilisables* par les novices.

CONTENTS

LIST OF FIGURES

1

INTRODUCTION

> *"It can scarcely be denied that the supreme goal of all theory is to make the irreducible basic elements as simple and as few as possible without having to surrender the adequate representation of a single datum of experience." (Albert Einstein, 10 June 1933, "On the Method of Theoretical Physics", The Herbert Spencer Lecture, Oxford – commonly **simplified** as "Things should be made as simple as possible, but not simpler")*

Creating and evaluating new interaction techniques is one of the main objectives of HCI research. Some techniques result from the incremental evolution of central branches of the field, others are punctual and isolated attempts to use either new technical possibilities or unexploited physical abilities to create alternative interaction methods. We, as HCI researchers, always need additional directions to frame the creation of usable and efficient new interaction techniques. Furthermore, with more and more complex computer systems, users need expressive interaction techniques, i.e., allowing them to communicate more information to the system in a single action, or to access a wider variety of elementary actions. As expressiveness will certainly make the interactive system more complex, designers must find ways to release the burden of users' investment in that complex interaction with the computer.

This dissertation explores how and why interaction designers can be inspired by expert activities and natural human abilities in sensemaking to design expressive interaction techniques while keeping them accessible to novices. We consider expert activities as activities in which some essential actions and underlying goals are only accessible to trained and skilled people. Such activities, e.g. wood carving or calligraphy, represent efficient collaboration of advanced human abilities developed through practice with adapted learning methods, and refined artifacts allowing experts to express themselves within the culture of the field. Our main hypothesis is that some elementary components of these activities are accessible to non-experts with little or no practice, and can be used for the design of expressive interaction techniques. These will not present the same complexity as the original activity, however will still allow for deep internalization and

improved expressiveness. Our central argument is that non-experts already have substantial knowledge from their own embodied experiences with their environment and the artifacts they use, and from their abilities to make sense of activities they observe. We take advantage of this knowledge to accelerate and simplify their access to expressiveness.

For all the projects presented in this dissertation as concrete explorations of our approach, music has been chosen as the source of inspiration. Both the experiences of musicians and non-musicians will be described. First, we will define the particularities of our position in the field of HCI.

1.1 CURRENT TRENDS IN INTUITIVE INTERACTION

One of the most surprising trends in HCI research is the idea that the relationship between human beings and computers may be absolutely intuitive. This would mean that users do not need prior knowledge or experience to interact with such *intuitive*, interactive systems. In addition, it implies that users will not need to engage actively in a learning process, since all they have to understand about the system is immediately ready-to-hand. As a result, there should not be any real difference between novices and experts when using such systems.

Figure 1: Clara Rockmore (1911-1998) and Lev Sergeyevich Termen (Léon Theremin, 1896-1993) playing the theremin

While ease-of-use is clearly an advantage for interactive systems, absolute intuitiveness seems to appeal to the same illusion as the immediate virtuosity supposedly accessible with some electronic musical instruments in the beginning of the twentieth century. Giving immediate access to the richness of music was one of the main commercial arguments for the Theremin (1929, Radiola Division, The RCA Theremin (advertisement)), grounded on the fact that playing this new instrument just required free in-air gestures, with no me-

chanical constraints such as the ones commonly imposed by acoustic instruments with keyboards, strings, membranes or vibrating tubes. But playing The Theremin is actually very difficult, since it does not provide any tactile or visual feedback to the player (see Fig. 1). If we look at the history of that so-called *intuitive* instrument, we discover that there were less than 10 expert players during the entire past century. Furthermore, seeing Clara Rockmore (1911-1998)[1] interpreting classical pieces on the Theremin gives some insights into the extreme difficulty she must have faced to reach such levels of precision and virtuosity which were often described as "imitating the human voice".

In a recent article, Don Norman addresses the question of the natural aspect of *Natural User Interfaces* (Wigdor and Wixon, 2011). One interesting idea presented in this article is that no system is "inherently more natural than others" (Norman, 2010b). He argues that the most important aspect for a system to be perceived as easy-to-use is its standardization allowing a stability in experiences. Keyboards and mice are not *natural*, but they have been refined over time and now they are standards every user feels comfortable with.

In *The Humane Interface*, Jef Raskin already presented a similar idea, by explaining that "there is no human faculty of intuition" (Raskin, 2000) (p. 150). Nothing is accessible with no prior learning process or experience and when users seem to act intuitively they just use methods and techniques they have acquired before. It is more accessible because it is familiar in some way. For the same reason, he also avoids the use of the word *natural*, commonly used today to qualify interfaces that do not require specific instructions. It clearly implies that there are some similarities between the use of these interfaces and other experiences of the subject, but that "naturalness", like "intuitiveness", remains difficult to define and quantify. Aside from that extreme definition, the term *natural* is often used to describe interfaces whose learning is quasi-instantaneous.

In fact, as early as in 1973, Douglas Engelbart presented the idea of a tradeoff between *efficiency* and *ease-of-use*. Even at this time, he already complained about this research for immediacy: "I believe that concern with the "easy-to-learn" aspect of user-oriented application systems has often been wrongly emphasized. [...] No one seriously expects a person to be able to learn how to operate an automobile [...] with little or no investment in learning and training." (Engelbart, 1973). This quote emphasizes the idea that ease-of-use will certainly depend on the complexity of the task. During the *Bootstrapping Seminars* in 1992, Engelbart insisted on the same idea by blaming researchers from the Artificial Intelligence community in the

1. See for example her interpretation of Saint-Saëns' *The Swan*: www.youtube.com/watch?v=pSzTPGlNa5U

70s for their will to create computers which can "adapt to the human [...] and not require the human to change or learn anything". In the light of the tradeoff he presented in the 70s, he observed that "it's sort of like making everything look like a clay tablet so you do not have to learn to use paper" (Bardini, 2000). Complex tasks could lead to richer experiences but they will definitely require more engagement from the user.

1.2 THE COMPLEXITY OF RICH EXPERIENCES

In his book published in 2010, Norman presents *Complexity* as the opposite of the aforementioned immediacy. Complex activities are not immediately accessible. In his book, he draws a fundamental separation between *complex* and *complicated* (Norman, 2010a). Human beings can deal with complex activities, i.e. activities with many intricate and interrelated parts, without experiencing confusion, which characterizes the experience with complicated systems: "I use the word complexity to describe a state of the world. The word complicated describes a state of mind" (ibid.). Therefore complexity is just an aspect of the rich and satisfying experiences we are seeking. People easily accept it if they see it as a necessary part of that kind of experiences.

Norman provides concrete examples to underscore that notion. The physical structure of a cluttered desk is complex but could be very clear for the person who owns the desk. Baseball rules are complex, but they are also an important part of the enjoyment of the game since fans and journalists like to debate the intricate rules and contradict the referees thanks to their detailed knowledge. Thoughts, stories, poetry and even knowledge could not exist without language and writing, which are two very complex activities. In these examples, the complexity of the activities is not only acceptable, it is also an essential part of the contemporary experience of living. We could make the same observation about several expert tools, e.g. pens or musical instruments, complexity of which is necessary in our culture. In fact, we often want experiences to be complex and we do not necessarily pay attention to the complexity of the activities we master. We do not remember the time we spent on learning them because of the satisfaction they provide us. In some cases, we might even call them *intuitive* once mastered after years of practice.

From the design point of view, complexity is not a problem and can be overcome as soon as it is not arbitrary (Norman, 2010a). The activity must be comprehensible by having a consistent internal logic, and the user should be guided through its understanding. Due to both the accountability of the activity and the user's investment in the learning process, the complex activity will never be perceived as confusing or frustrating.

1.3 EXPERTISE AS AN INSPIRATION

Expert activities are inevitably complex regarding the definition given in the previous section. Furthermore, Norman proposes the use of the time required to learn an activity as a measure of its complexity (Norman, 2010a). With that sole measure, the complexity of expert activities seems undeniable. But an important advantage of expert activities is that experts share knowledge and tools which make them able to overcome complexity. Expert artifacts, both physical and psychological, have been progressively refined in a "*co-adaptation*" (Mackay, 1990) between experts and their activity. Using these artifacts becomes automatic with practice, allowing experts to efficiently reach their goals without being constantly aware of the all-comprehensive complexity of their activity. Furthermore, expert artifacts act as standards according to Norman's definition (Norman, 2010b) since whole expert communities use them in their activities. Expert activities are thus based on fine-tuned associations of human motor, perceptive and cognitive abilities, refined artifacts, and efficient practice and learning methods. In our search for expressive interaction techniques, expert activities are good sources of inspiration, being the best examples of situations in which human beings efficiently cope with high levels of complexity.

But as we want our interaction techniques to be accessible to novices, we will establish a two-way relationship between expert activities and novice abilities. From expert activities, we can take some isolated components and efficient learning methods if needed. And on the novices' side, we can observe and reuse the knowledge and skills they already have and their understanding of expert activities. In fact, novices can have sizeable knowledge of some expert activities, gained through a certain engagement as perceivers. Artistic activities perfectly fit that consideration since audiences actually participate in the activity by perceiving, understanding and appreciating artistic expression. Novices even develop particular perceptive and motor skills from their exposure to artistic activities.

This approach will help us maximize Engelbart's tradeoff between *efficiency* and *ease-of-use* (Engelbart, 1973). We identify some requiments to make expressive interaction techniques accessible to novices:

- Reduced complexity: Resulting interaction techniques must be far less complex than the original expert activity;
- Usability: Interaction techniques must take into account the innate abilities and skills of users;
- Learnability: Learning methods must be provided, so that novices can become experts;
- Evaluation: We must propose a framework to help evaluate usability with standard HCI evaluation methods (e.g., controlled experiments, field studies).

Our implementation of this approach focuses on the expert activity of playing music. The next chapter presents approaches to human experience and activity that are relevant to our work. This background introduces the concepts we will use in the rest of the dissertation to describe human skills, and allows us to identify what is missing in interation design from that point of view.

The third chapter analyzes inate abilities for social understanding allowing human beings to make sense of some expert activities in which they are engaged, such as music. We analyze musical expertise and the music-related skills developed by non-musicians. We then provide a framework to create expressive interactive systems that are accessible to novices.

The next three chapters present the three main projects of this dissertation. Each of them is inspired by music playing and takes advantage, to varying extents, of novices' knowledge and understanding. The first one addresses the use of rhythmic patterns as an input method to trigger commands, and an exploration of efficient feedback mechanisms. The second one addresses the design of chording gestures for multitouch displays and proposes an adapted learning method inspired by the teaching of chords in music. The final project presents a functional decomposition of acoustic instruments from which we draw a framework for improving the "instrumentality" of computer music systems.

In the final chapter, we will provide insights on a few additional explorations available with our approach.

The first contribution provided in this dissertation is the framework introduced in the third chapter to help designers create expressive and accessible interaction techniques inspired by expert activities. It focuses on accounting for the embodied and social aspects of human experiences, as well as for the knowledge and skills of non-experts. This framework provides additional descriptive, evaluative and generative powers to Instrumental Interaction (Beaudouin-Lafon, 2000) and the Human-Artifact Model (Bødker and Klokmose, 2011) by considering the relationship between the functioning of instruments and the innate skills, available knowledge and abilities in sense-making of common users. Each of the projects presented in the fourth, fifth and sixth chapters present various contributions to specific fields of application in HCI.

2

PHILOSOPHICAL AND PSYCHOLOGICAL BACKGROUND

"Living in a material world
And I am a material girl
You know that we are living in a material world
And I am a material girl"
(Madonna, 1984, "Material Girl" on the album "Like a virgin", Sire Records and Warner Bros Records)

In this chapter we will present the concepts and the vocabulary we will use for the description of human experiences and skills. The questions we attempt to address are: What are the fundamental aspects of engaging in an activity? How can we describe the involvement of an expert? How does one deal with the knowledge of an expert activity?

In all expert activities, becoming an expert implies gaining the ability to use complex technical artifacts and concepts. We will first present theories that distinctly consider the knowledge which is in the environment and the artifacts we use, i.e. *what is in the world*, from purely intellectual knowledge, i.e. *what is in the mind*. But some other approaches go further and define the interaction and the mutual influence between humans and their environment as the central part of experiences. Both phenomenology and Activity Theory consider human activities as mediated through the body and various artifacts, and highlight the social aspects of human sense-making. These two approaches have also spawned advanced HCI frameworks, such as Instrumental Interaction (Beaudouin-Lafon, 2000) and the Human-Artifact Model (Bødker and Klokmose, 2011).

2.1 REMAINS OF MIND-BODY DUALISM

2.1.1 *Affordances*

Most of the work of James J. Gibson (1904-1979) has focused on how we understand our environment through visual perception. After much work on his ecological approach to visual perception, he introduced the term *affordance* in 1966 to describe action-relevant prop-

erties of the environment that are accessible through vision (Gibson, 1966). In other words, the environment provides more information at first glance than just its physical properties. We also get clues as to what actions are possible to apply to physical artifacts. For example, a plain surface of wood affords sitting and a hand-sized object affords grabbing. Affordances refer to the meaning artifacts directly provide to the observers, with no additional understanding and no prior knowledge required. This meaning is not supposed to be adapted to the observer's goals and affordances do not necessarily give clues as to the correct use of the artifacts.

While this theory is supposed to be applicable to the visual perception of any animal, the main critique against Gibson's definition is that action possibilities always depend on the observer's needs, which could evolve over time. For me, my computer's keyboard affords typing. However, for a cat, it may afford walking on. Therefore, the "objective" information obtained through visual perception is not sufficient to lead action. For expert activities such as music, such affordances do not provide any proper information. It seems obvious that people need training to know how to operate a trumpet properly. Its shape does not afford anything else than grabbing it, pushing the valves and eventually blowing in the mouthpiece which will not produce any sound.

The first interpretation of the concept of affordance adapted to interaction design is provided by Norman in the late 1980s. According to him, affordances not only give clues as to the possible actions we can do on an artifact, but also inform *how* to operate it. For example, a doorknob affords turning in addition to grabbing. From the design point of view, Norman argues that these *perceived affordances* are accessible through *signifiers* that can be created by designers to inform proper use : "A signifier is any perceivable sign of appropriate behavior, whether intentional or non-intentional" (Norman, 2010a, p.227). While this precision tends to clarify the use of affordances in interaction design, this augmented definition still pays little attention to what happens on the observer's side. Norman himself insists that understanding is important in perceiving our environment, while the actual processes of sense-making and learning are not addressed in his book. However, he gave a central role to *conceptual models* or *mental models*, which reside in people's minds and guide their interaction with the environment (ibid. pp.247,253).

2.1.2 *Mental Models*

For Norman, a *mental model* "helps us transform complex physical reality into workable, understandable mental concepts. [...] They enable us understand things, learn how they work, and figure out what to do when failures occur." (Norman, 2010a, p.37). Fred Ler-

dahl also holds true to this principle in his cognitive approach to music. He stated that "comprehension takes place when the perceiver is able to assign a precise mental representation to what is perceived" (Lerdahl, 1988, p.232). Alan Blackwell defines mental models as "objective, formalizable, and symbolic entities" encapsulating the user's understanding of the system (Blackwell, 2006). Card and Moran have largely investigated the application of that notion to HCI with their early *Model Human Processor* (Card and Moran, 1986). They support the idea that the user's mental model corresponding to a computer system should include knowledge of where data is stored and how it is manipulated, in order to make him able to understand its behavior and predict the outcomes it will provide when commands are executed.

Even if their ideas have participated in the evolution of the desktop metaphor which is the most widespread type of user interface, new electronic devices do not provide this kind of information to the user. However, they are still usable. For example, only few experts know about the structure of the iPhone file system. In music also, this requirement does not seem to be the core of the problem since precise knowledge of the mechanical structure and functioning of an instrument, i.e. *how it works*, is not necessary to play it. For example, a pianist can play piano by focusing on its keyboard, without knowing much about the internal mechanisms of hammers, dampers and strings. Thus we are still in the position of wondering how pianists acquire the knowledge of how to play the piano, even if "a system has to be designed with an explicit conceptual model that is easy enough for the user to learn" (Card and Moran, 1986). How do they learn it ? While reminding interaction designers that the user's understanding of the system is a central matter, this approach fails at describing the process of sense-making and at characterizing the nature of expert knowledge and skills.

From our everyday observations, expertise seems to be more than decoding information provided passively by physical structures of artifacts and collecting large amounts of universal abstract representations of the reality. In any case, all of these approaches, either claiming that knowledge is in the artifact or that it is in the user's mind, seem insufficient to describe the dynamic relationship between people and the world. How do they know how to use artifacts to fulfill their evolving goals? How do they make different decisions in the same situation depending on their needs? How do they reuse prior knowledge in new complex situations? How do they learn from the experiences of other people? And in particular for our main investigation: What knowledge and skills do they develop when they become experts?

We will now present two approaches focusing on the definition of that particular relationship between humans and their environment: Embodiment and Activity Theory. Each of these has different bases, but they will help us understand how humans make sense of their environment, learn and become experts. They are both grounded in the idea that all knowledge emerges during bodily and artifact-mediated actions constituting subjective experiences. They maintain that there is a fundamental unity of the mind and the world, and give an important place to the social nature of human experience.

2.2 EMBODIMENT AND ENACTION: BODILY AND SOCIAL EXPERIENCES

An important review of phenomenology can be found in Dourish's book on Embodied Interaction (Dourish, 2001).

2.2.1 *Phenomenology*

Embodiment and enaction take their roots in the phenomenological tradition founded by Edmund Husserl (1859-1938). Phenomenology studies the way we perceive and act in the world in which we live, with a central emphasis on the phenomena of experience. It first rejects the Cartesian dualism which defines a separation between mind and body. Contrary to the aforementioned approaches, phenomenology neither considers that knowledge is made of mental abstractions stored in our memory, nor that it is directly accessible in the world. Initially, the main idea was to establish a parallel between the objects of perception and the acts of perception. Both are considered as phenomena. The structure of these mental phenomena, including consciousness and subjective experience, became the focus of Husserl's studies.

Martin Heidegger, one of Husserl's students, rejected this exclusive focus on subjective experience separately from the study of the world. He defends the idea that the fact of being and our environment are inseparable, with his concept of "being-in-the-world" (Heidegger, 1927). In fact, thinking does not occur separately from being and acting, and the world we perceive exists through our perception of it. Our existence resides alongside the existence of our environment, and takes place in our very interaction with it. Parts of that environment could be used to fulfill our needs and become *equipment* in Heidegger's terms. This equipment is defined as something we act through in order to achieve our goals. The dual nature of *equipment*, being both something we act through and something that exists in and of itself as a part of our environment, gives rise to the definitions of *ready-to-hand* and *present-at-hand*. When I play the guitar, I act *through* the instrument to play music. It disappears from my im-

mediate concerns and can be considered as an accessible background of my main activity, i.e. music playing. It is *ready-to-hand*. But it can sometimes be the object of my attention, i.e. *present-at-hand*. At these moments, for example if I need to change a broken string, it exists for me as an entity of interest.

Alfred Schutz added a social dimension to phenomenology in the early 1930s. His concept of *intersubjectivity* (Schutz et al., 1932, tr. 1967) focuses on the fact that we establish a relationship between our own experiences and those of others, which allows us to build a common framework for meaning. In this two-way relationship, we understand how our activities are perceived by others, and we are able to interpret their activities by grasping their intentions and goals when observing their actions. That way, we access a part of their experiences from our point of view, and learn from it implicitly. We will come back to that notion in the following chapter as a motivation to provide interaction techniques for novices based on the intersubjective knowledge they have of expert activities.

2.2.2 *Embodiment And Knowledge Reutilization*

The concept of *embodiment* became central in Merleau-Ponty's PhD thesis in 1945 (see Merleau-Ponty, 1945). In his work, embodiment describes the body as the mediator between the mental phenomena studied by Husserl and our relationship with the world explored by Heidegger, and consequently between the subject and the object. He stated that our embodied nature is inherent to both our internal and external experiences. Therefore, embodied action, i.e. *enaction* in Varela's terms (see Varela et al., 1991), is where meaning emerges. The reality of our bodily experience and the perceived existence of our environment are co-constructed during enaction and are not accessible beforehand as abstract representations. Embodiment also has consequences for the definition of perception, which becomes a perceptually leaded action to enact some meaning from our environment and the actions of others, rather than a contemplative and passive behavior. Gibson's and Norman's affordances would not provide any information if we did not have this embodied relation with our environment.

In 1996, Dreyfus proposed a differentiation between three distinct definitions of embodiment in Merleau-Ponty's work. The first is the physical embodiment, i.e. the innate structure of our body in the world. The second is the set of basic common skills that are available with our bodies. And the third is the set of cultural skills we acquire from the interaction with the world in which we are embedded (Dreyfus, 1996). This observation paves the way for the study of skill acquisition in the light of embodiment. In his paper, Dreyfus

describes skill acquisition by instruction and discriminates among different levels of expertise:

- For a *novice* subject, the instructor decomposes the situation into elementary context-free *features*. The *beginner* can recognize these features and respond to them with actions leaded by some very simple rules, without ever having prior exposure to the actual task. In the example given by Dreyfus of the person learning to drive, the novice driver concentrates on features like the speedometer needle to know when to shift gears.
- An *advanced beginner* knows these elementary features and is also able to note additional meaningful *aspects* that he discovers through actual experience. In Dreyfus' example, the advanced beginner is able to know when to shift gears by listening to the sound of the engine, interpret the speed and trajectories of other vehicles and predict the behavior of pedestrians. He is now able to understand the situation more deeply.
- In order to make decisions in new situations, he will perceive features and aspects in a hierarchical way. In fact, the possible situations are far more numerous than the few features described by rules for the beginner. At this stage, the subject is considered as *competent*. He is now able to make subjective choices that are beyond the elementary instructions he has received, and he will experience good or bad feelings depending on the result of his choices. He will modify his response to the situation if it does not produce a satisfactory result, entering the *feedback loop*.
- Then the subject becomes *proficient* when he makes more and more instantaneous situational discriminations. His knowledge allows him to avoid the conscious successive observation of features and aspects, and provides him with the capability to react properly to new complex situations. He is able to immediately identify salient aspects which are related to his goal. The analysis of the situation is now immediate, he sees what needs to be done, but he still has to consciously decide how to do it properly.
- In the end, the subject becomes an *expert* when he is able to make very subtle and refined discriminations among situations while knowing immediately how to act to achieve his goals. His perception of the situations is highly structured, and performing the appropriate action is now as immediate as understanding the situation. Situations are no longer the objects of attention. "This allows the immediate intuitive situational response that is characteristic of expertise. [...] What must be done, simply is done".

According to that definition of expertise, an expert acts by instantaneously reacting to a given situation, without referring to an internal *representation* of his knowledge and without requiring purely intellectual processes. Merleau-Ponty refers to this state as "skillful coping".

Perception is accurate, and responses to situations have been sculpted by experiences and are efficient and adapted to the subject's body and to his goals: "The body takes over and does the rest outside the range of consciousness" (Dreyfus, 1996).

One question that needs to be asked when considering that we do not refer to internal representations when acting, is about our ability to generalize situations. In fact, it is impossible to consider every situation as unique and different from all the others. We must immediately identify similarities between these in order to reuse our skills and knowledge. For example, when I play the piano in a recording studio, I use the same knowledge and skills as when I play mine. Merleau-Ponty proposes three constraints to the generalization of situational knowledge. The first is due to the brain architecture, with its inherent capacities and limitations. The second relates to the aspects of *supervised learning*: the order and frequencies of situations' occurrences during skill acquisition. The last constraint is the reinforcement provided by the resulting satisfaction when coping with situations.

The way we generalize situations and observe similarities, and in the end the way we understand situations that arise, relies on these three constraints leading to what Merleau-Ponty calls the *sedimentation* of experiences. These constraints influence how the subject identifies an action as the optimal response to a situation, and further situations will be perceived relatively to this optimum, according to what Merleau-Ponty calls the "maximum grip" that a subject aims to have on the situations he faces. Therefore, the world we perceive is the best possible configuration allowing that maximum grip given these constraints.

While insisting on the fact that the construction of expert knowledge is not made of storing representations of situations in memory, Merleau-Ponty does not concretely explain what is sedimentation made of. A possible explanation comes from the field of Artificial Intelligence and is reported by Dreyfus in his article (Dreyfus, 1996): reaction could rely on the architecture of neuron networks, constantly reorganized by experiences.

2.2.3 *Direct Implications For HCI*

In his book on embodied interaction, Dourish underlines some directions for interaction design inspired by phenomenology and embodiment. First, computational *abstractions* are necessary in interface design. By hiding the implementation part, abstractions make it possible for us to deal with complex computational behaviors through simpler high-level objects. They simplify the situation users have to tackle, supporting their ability to identify interesting aspects. They

will also help users generalize situations. For example, the abstraction of data provided by file systems avoids coping with low-level data, and allows users to perceive it in an appropriate manner regarding their needs: as separated files which could be opened with various applications on various devices. Even if mobile operating systems are radically changing the way we deal with data as files, every graphical user interface based on the desktop metaphor still provides file systems that users are able to understand and manipulate.

Second, *accountability* is an important aspect of interactive systems. It is the way an interface gives an account of what is happening, as our physical environment does. For Dourish, it is "an explication of how the system's current configuration is a response to the sequence of actions that has led up to this moment, and a step on the path toward completing the larger action in which it is engaged" (Dourish, 2001). In HCI, it is often addressed by implementing mechanisms such as *feedforward* and *feedback*. The latter consists of showing what actions have been taken into account by the system and what are their results, while the former shows what actions are available and eventually how to perform them and what their consequences would be. Unlike affordances, these two mechanisms support accountability and therefore guide the user in understanding the system and knowing how to operate it.

What we learn from the concept of embodiment, given the focus of our study, is that we cannot consider experience separately from our bodily presence in the world and our nature as social beings. Expertise cannot be regarded as the collection of abstract mental models. It is characterized by one's ability to organize perception and action to provide instantaneous analysis of the situation, decision and response corresponding to one's goals.

The notion of equipment insists on the "readiness-to-hand" of mastered tools. In expert activities, the equipment is no longer the object of attention, but rather an assisting entity to enrich the natural abilities of the subject. That is why expert musicians often describe their instruments as extensions of their bodies.

Perceptive and motor skills are determined by the intrinsic limitations of our bodies, the actions we can perform with it, and the knowledge we have acquired culturally and socially. From the point of view of novices observing expert activities, we have seen that even perception is an active embodied process aiming to enact meaning. We have presented the concept of intersubjectivity supporting the idea that part of our knowledge and experiences are shared through simple observation. In the next chapter, we will see how musical activities are experienced by non-musicians.

When designing interactive systems, abstractions will help all users, including novices, to understand and generalize situations. Accountability will participate in sense-making and skill acquisition, by improving the perceived consistency of the system.

The concept of skillful coping used to describe expertise is somewhat different from the *internalization* and *functional organs* proposed by Activity Theory and presented in the next section. From this latter point of view, the ability to abstract experienced situations and internalize more and more sophisticated actions plays an important role in expertise.

2.3 ACTIVITY THEORY: ARTIFACTS AND KNOWLEDGE

A very complete overview of Activity Theory is provided in the recent book on Activity Theory and HCI by Victor Kaptelinin and Bonnie Nardi (Kaptelinin and Nardi, 2012).

2.3.1 *The Structure Of Activities*

In this book, the authors identify many similarities between phenomenology and Activity Theory: both argue for a unity of the mind and the world, and provide an appropriate description of subjective experiences. Apparently, the main difference is the philosophical origin of phenomenology, while Activity Theory comes from psychology. For our study of expertise, an interesting difference is that embodiment seems to focus on how experts *react* to situations given their *implicit* goals, whereas Activity Theory tends to describe how experts *act* in various situations to fulfill their *conscious* goals.

Activity theory was first introduced by a group of Russian psychologists, including Lev Vytgotsky (1896 – 1934) and Aleksei N. Leontiev (1903 - 1979), after the Russian revolution of 1917. Their theoretical framework was based on three main concepts to describe the emergence of meaning and the interaction of human beings with their environment: tool mediation, the concepts of internalization and externalization, and the social nature of humans (see Vygotsky and Cole, 1978). Activity is the relationship between the subject and his object, and the theory tends to analyze its construction, structure, and processes depending on the cultural and social contexts. This relationship is grounded on two elementary rules: subjects have *needs* or *goals*, and subjects and objects determine one another during activity. This latter claim was also supported by phenomenology, and leads to "the principle of unity and inseparability of consciousness and activity" formulated in 1948 by Sergey L. Rubinstein (1889-1960). Like embodied action, it links the internal and external phenomena, i.e. human consciousness and human actions in the world, which mutu-

ally influence one another. Contrary to cognitive approaches, Activity Theory focuses on the actual manifestations of activities, with their particularities and situated aspects.

An activity is guided by a motive that meets a certain physical or psychological need. Physical needs depend on what is required by our body to live, while psychological needs are influenced by culture and social life. A need is the direction of the activity, and the motive is what stimulates the subject, i.e. the motor of the whole activity. Subjects are not necessarily aware of their motives, nor are they necessarily aware of their psychological needs.

Two constitutive elements of activities are described in a hierarchical manner: *actions* and *operations*. Actions are the different steps required by the activity. They may not be directly oriented towards the motive, but are oriented to achieve conscious subgoals. Operations are other units of activity through which the user can realize the desired actions. They are automatic and performed unconsciously by the subject. When performing an action in a situation which is slightly different from the usual one, operations can emerge as an *improvisation*, i.e. an adjustment of an action to the given situation. Operations could also come from the practice of the same action in similar situations, as an *automatization* of the action. Operations, in their turn, can become actions if they do not provide the desired outcome and need to be consciously considered and modified.

For example, one of the motives of a piano maker is to build pianos. Some of the actions he will have to do are buying wood, transporting it, cutting it, and assembling the parts of the piano. Making phone calls to choose a wood provider is not directly related to the actual construction of the piano, but it is a necessary action to fulfill the overall need. Examples of operations done by the piano maker could be using his saw to cut wood in his workshop, i.e. *automatized* actions, or nailing some pieces down with an alternative tool if he breaks his hammer, i.e. *improvised* actions.

At this point, it is worth noting that the definition of *plans* and *situated actions* provided by Suchman (1987) in ethnomethodology seems complementary to the approach to actions and operations provided by Activity Theory. Suchman observed that people generalize the actions they perform as *plans*. Within Activity Theory, this process relates to the definition of *internalization*. Then, the subjects execute their plans as *situated actions*, which necessarily depend on the conditions of each particular situation. If some conditions disrupt the execution of the plan, actions must be improvised, and replanning is necessary. Therefore, she proposes to study these processes to understand how people consider the particularities of the situations and abstract the actions they perform, in a flexible enough manner to reuse knowledge in further situations. Her contributions, emphasiz-

ing the importance of the situational conditions in the planning and execution of actions, have been largely used in the field of HCI and has led to design methods such as Participatory Design (Greenbaum and Kyng, 1991), in which interaction designers observe work tasks within their context and cooperate with users.

2.3.2 *Mediating Artifacts And Embedded Knowledge*

After this general definition of Activity Theory, we will describe its two main concepts that are useful for our study of expertise. First, in most cases, human activity is mediated by artifacts. This is one of the main differences between humans and other animals. We not only use technical tools, e.g. hammers or pliers, to affect physical objects, but we also use languages, mathematics, physics, maps, and various other symbolic systems to interact with our understanding of the environment and communicate with others. These kinds of artifacts are called *psychological artifacts* by Vytgotsky (Vygotsky and Cole, 1978). For example, a pianist playing Western tonal music will also use harmony, tonality and rhythm as artifacts to mediate his activity of playing music. From that perspective, mental processes become instrumental acts too and internal activity is not grounded on mental models but mediated through psychological artifacts.

One of the main aspects of the central place of artifacts in Activity Theory lies in the fact that they are considered to be more than static entities used during an activity. In fact, artifacts, human actions and understanding have an influence on one another. On the one hand, the actual shape of the piano has been refined over 600 years in line with the evolving ways of using it to play music. On the other hand, piano players and guitar players will have very different physical relations with their instruments, and may even have different visions of chords and harmony in general. Therefore, the structure of the tool and the acquired expertise to mediate activities through it impact on how humans create meaning. An artifact could be seen as an accumulation of social and cultural knowledge relative to the activity, and implicitly participates in its transmission. Indeed, an expert artifact used in an activity has allowed all experts to fulfill their needs.

As illustrated by Bødker and Klokmose in a recent article "The western forks and knives embody in their design both the possibility of cutting and lifting certain shapes and textures of food, and an assumption of the western traditions of how to eat properly" (Bødker and Klokmose, 2011). In the same article, the authors present an idea similar to Heidegger's concepts of *ready-to-hand* and *present-at-hand*. They state that "a mediator that works well allows the user to focus on the object of interest when carrying out the necessary acts supported by the capacities of the mediator. A mediator that does not work well causes breakdowns and draws the user's focus towards the

artifact as such". Then, the concept of *functional organs* from Leontiev is proposed to describe the quality of the mediation (see Leontyev, 1981). Functional organs are created through the appropriate combination of human capabilities and artifacts. To create a functional organ, one will need to know precisely what outcome the artifact can provide, higher-level goals achievable with it, how to operate it physically, and how to use it to fulfill particular goals (Bødker and Klokmose, 2011). Kaptelinin and Nardi propose the distinction of two kinds of affordances to take these various aspects of mediation: *handling affordances* show how to operate the artifacts, and *effecter affordances* relates to the possibilities for employing the artifact to make an effect on an object (Kaptelinin and Nardi, 2012). Appropriating an artifact to such a level that it becomes a functional organ allows a deeper engagement and improves efficiency, which is certainly an aspect of expertise.

2.3.3 *Internalization And Externalization*

Another core idea of Activity Theory lies in the concepts of *internalization* and *externalization*. Internal activities correspond to what we usually call mental processes. The theory defends the idea that internal and external processes cannot be analyzed separately, because the former can only be created through the internalization process. Internal activities allow the subject to interact with his understanding of the environment without performing physical actions, but just by simulating it mentally. It provides him the power of imagination, mental combination and planning, which are expert skills that are especially important in creative activities. The system of numeration provides a good illustration of internalization. When an infant learns how to enumerate things, he first needs to count with physical objects. Then, he becomes able to count with his fingers. Finally, he is able to count mentally, without requiring external activities. The process is now mediated only by newly created psychological artifacts.

Given that focus on artifacts and mediation, Activity Theory has been used in HCI to describe the computer as a mediator to act upon objects or with other subjects. The mutual influence of artifacts and subjects in the theory provides an interesting point of view for studying how the introduction of new computer systems can change practices, and how activities sculpt these systems.

This short overview of Activity Theory promotes the fact that activities rely on actions directed to conscious goals, and unconscious operations. In new situations, operations can be improvised to implement actions. With practice, more and more operations are created by automatizing actions. It justifies why experts do not need to con-

centrate to perform every single operation of their activity, but rather they perform it automatically.

Most human activities are mediated through artifacts. Artifacts are not only technical but also psychological. In music, as in language, many artifacts are of that second kind, such as rhythm or harmony. In the next chapter, we study the particularity of such artistic activities performed in front of the audience: even the potentially novice audience uses psychological artifacts in their activity of extracting structure and meaning from art pieces. In fact, listening to music familiarizes us deeply with the language of music. That is why so many people have strong musical tastes without practicing music, and why music lovers are able to appreciate fine subtleties between different musical pieces or even between different performances of the same piece.

The structure of the perception and action of both performers and listeners is altered by the technical and psychological artifacts used in the activity. Therefore, we can say that the perception of non-musicians attending concerts is sculpted by the rules of music. And on the other hand musical instruments and associated gestures depend on musical culture and the abilities they provide to communicate with non-musicians through the language of music.

Expert artifacts could be considered as functional organs as defined by Activity Theory. When using them as mediators, experts know precisely what actions the artifact can handle, how to use it, the results they can expect from their actions and the goals they can attain. To allow such a relationship with experts, artifacts must present *handling affordances* to inform their manipulation, and *effecter affordances* providing clues as to what outcome they can provide.

Practice allows subjects to internalize processes. Thanks to internalization, they become able to imagine, plan and predict the results of their actions. In music, as we consider listening an active process, we can say that listeners internalize some parts of the musical activity and are able to process it internally.

Even if the approaches presented in the two last sections do not provide many all-inclusive methods to design interactive systems, some frameworks have been drawn in the past two decades to enrich our view of interaction design in light of these new considerations on the relationships between humans, their environment and their instruments.

2.4 HCI FRAMEWORKS BASED ON MEDIATION

The philosophical and psychological theories presented in the previous sections played an important role in the emergence of the third

wave HCI, which considers interaction as a way to fulfill goals by performing situated actions, and to create meaning from the embodied use of computers. This definition tries to embrace all the facets of using the computer as a complex mediator.

While these approaches have been criticized as being difficult to apply to concrete design situations (see Harrison et al., 2007), they inspired several interaction models –i.e. sets of principles, rules and properties– aiming to help designers make good use of the rich relationship between artifacts and human action and understanding, such as Instrumental Interaction (Beaudouin-Lafon, 2000) and the Human-Artifact Model (Bødker and Klokmose, 2011). In the following, we focus on the interaction models which are centered on the notion of instruments, since expert activities are mostly mediated.

2.4.1 *Instruments And Schemes*

Many activity theorists progressively introduced a specific definition for *mediating instruments*. Pierre Rabardel and his colleagues proposed a distinction between artifacts and instruments, whether they are technical or symbolic. An instrument requires the subject to adapt his activity to its functioning by implementing *utilization schemes*, and will be sculpted by the activity in return. This dual process is called *instrumental genesis* (Rabardel, 1995). Thus, the instrument has two components: the first one resides in its nature as an artifact, and the second one is the human part of its nature, related to the defined utilization schemes which are known by the users. As it was the case for functional organs, utilization schemes define how to operate the instrument, what are the possible outcomes, which are the available goals and which are the ways to reach these. Schemes can be directed to the instrument itself, to the object of interest or to other subjects.

Utilization schemes are generalized thanks to *assimilation*, i.e. the possibility to integrate the use of several artifacts in one scheme to generalize it, and *accommodation*, which resides in the modification and combination of previously elaborated schemes. This notion of utilization schemes draws on Piaget's definition of *action-schemes* in his work on skill acquisition (Piaget, 1973). In Piaget's work, assimilation consists in integrating new experiences to schemes of knowledge in order to enrich them, and accommodation transforms schemes to adapt them to new situations. Therefore, an utilization scheme is an active structure in which previous experiences and knowledge are incorporated and organized. Within that definition, a single artifact could be seen as a collection of instruments, depending on the utilization schemes used by the subject. Thus, a hammer can be used as a *hammer* instrument in order to nail, or as a *weapon* or a *bottle opener* depending on the schemes used in various situations. In addition, it provides Activity Theory with two aspects that embodiment

is lacking: experts store their knowledge as action-related utilization schemes, and they generalize their instrumental knowledge thanks to accommodation and assimilation.

Artifacts are gradually instrumentalized through activity, which requires the intensive participation of subjects to merge the instrument with their activity (Béguin and Rabardel, 2001). This approach puts the instrumental genesis in the center of activities, which is adapted to cooperative design methods involving users and designers, such as Participatory Design (Greenbaum and Kyng, 1991).

2.4.2 *Instrumental Interaction*

While this definition of instruments has been mainly used for studying pedagogy, Beaudouin-Lafon introduced the concept of *Instrumental Interaction* to the field of HCI (Beaudouin-Lafon, 2000). Compared to the definition provided by Activity Theory, instruments do not necessarily present aspects of expertise, neither in their structure nor in their use.

Three of the contributions provided by Instrumental Interaction are particularly interesting for our approach. First, Instrumental Interaction provides an interesting review of the standard model of Graphical User Interfaces (GUIs). Second, it provides a definition for instruments in HCI and describes some properties on which they can be compared. Lastly, it defines three aspects among which to evaluate and validate interaction models.

Instrumental Interaction is inspired by our everyday use of technical instruments to mediate our activities. The idea is to define an interaction model to allow the study of any computer system, from the level of the whole system to the level of a single element of a GUI, as an instrument. This interaction model aims to inform the design of computer interfaces and stands as an extension and an operationalization of Direct Manipulation introduced by Ben Shneiderman (Shneiderman, 1983). The study starts from a review of the standard interaction model used for the design of GUIs: WIMP (Windows, Icons, Menus, Pointers). Even though this model was fundamental for the standardization of user interfaces and subsequently to allow broader access to computers, its limits are exposed by Beaudouin-Lafon: pop-up menus and dialog boxes are modal and force the user to switch focus, the commands that are accessible through menus and dialog boxes are too numerous, WIMP GUIs require screen space and increase visual complexity, and interaction with WIMP interfaces is mostly based on discrete pointing. He insists on the fact that the exploration of available commands becomes cumbersome using these hierarchically hidden ways to discover how to interact with the computer.

As a model, Instrumental Interaction provides the definition of some useful aspects for comparative analysis of interaction techniques as instruments. An instrument is a mediator between the user and objects of interest. The user performs a physical action to control the instrument, which sends a command to the object. In return, the instrument might provide feedback about the object's responses, in addition to manifesting its own reaction, through any available output modality, e.g., tactile, visual or auditory. An instrument is the combination of a physical part (input and output devices) and a logical part (representation of the instrument that reacts to some actions and controls the object of interest). As an example, Beaudouin-Lafon describes a scrollbar, that requires a mouse and an on-screen scrollbar widget (Beaudouin-Lafon, 2004). The combination of the two allows scrolling and provides feedback both on the widget and on the graphical representation of the scrolled document. In its definition, Instrumental Interaction clearly inherits some characteristics from Activity Theory, but instrument design is also inspired by the theory of Situated Action (Suchman, 1987) since the examples provided in the articles are based on the observations of users' activities in the actual context of use.

Instruments can be described and compared according to the following properties (Beaudouin-Lafon, 2004):

- The activation of the instrument, i.e. how to select the instrument to use it in subsequent actions. For a scrollbar, the activation is spatial. I can use it only if I click on it and drag it. Whereas, the rectangle creation instrument in drawing applications is activated temporally. I have to first click on it and then I can use it, until I activate another instrument;
- The degree of indirection composed of two aspects: the spatial offset is the on-screen distance between the graphical part of the instrument manipulated by the user, and the graphical representation of the object controlled through the instrument; the temporal offset, which is the delay between the action and the modification of the object;
- The degree of integration measures the degrees of freedom of the input device that are used by the instrument;
- The degree of conformance measures the consistency between the physical actions of the user and the modification of the object.

Beaudouin-Lafon and Mackay defend the idea that Instrumental Interaction supports three processes they define as necessary for the standardization of interaction techniques: *reification*, *polymorphism* and *reuse* (Beaudouin-Lafon and Mackay, 2000). From the point of view of Activity Theory, *reification* is the externalization of processes to shape instruments. It corresponds to Rabardel's instrumental genesis but, in HCI, instruments could be physical, digital, and mostly both. The physical and logical parts of instruments can be *polymor-*

phic, i.e. used to manipulate different objects: for example, a color palette could be used to change either the color of a text or the color of a shape in a drawing application. Polymorphism is like considering direct assimilation and accommodation of utilization schemes in Rabardel's term. *Reuse* is the act of re-applying previous input or output, like macros in spreadsheet applications or style sheets in word processing software. Thanks to the standardization of reified instruments through the improvement of their polymorphism, reuse is facilitated and instruments can be shared more easily. Together, these three principles allow to reduce the number of instruments we use and increasing their efficiency.

As a validation of Instrumental Interaction, the authors show that existing interaction techniques could be described and analyzed using this model. They also show that Instrumental Interaction allows for the design of innovative and interesting interaction techniques and then they propose appropriate dimensions for the study of interaction models (Beaudouin-Lafon, 2000):

- *Descriptive power*: the ability to describe a significant range of existing interactions
- *Evaluative power*: the ability to compare alternative designs and choose among them depending on the situation
- *Generative power*: the ability to help designers create new designs

2.4.3 *The Human-Artifact Model*

For Bødker and Klokmose, Instrumental Interaction points "towards an understanding of human-computer interaction, where instruments coexist and replace each other as extensions of the human body" (Bødker and Klokmose, 2011). In the same article, they propose an extension of Instrumental Interaction by reintroducing the culture / experience / practice triptyque, and the notion of artifacts ecologies: the *Human-Artifact Model*. The framework they draw aims to analyze and compare instruments by answering the three questions, "what", "why" and "how" to characterize the levels of mediated activity. In their analysis, they respectively address these questions by describing the *instrumental* level, i.e. which activity can be done with the artifact, the *motivational* level, i.e. which motives could be supported by the artifact, and the *operational* level, i.e. how it is operated and which actions can be done with it. Studying eventual breakdowns in these levels of activity allows to compare an artifact to the ideal concept of functional organ. Given these definitions, the properties defined by Instrumental Interaction refer to the relationship between the operational and the instrumental part, i.e. the relation between the behavior of the instrument and the objects of interest.

In summary, the nature of an instrument comes from the co-adaptation (Mackay, 1990) between the subject, the artifact and the activity. Instrumental genesis, i.e. the emergence of a functional organ, are potentially long processes which require the subject to learn the technical and functional aspects of the artifact as a mediator to reach his goals. In order to make expressive interaction techniques immediately usable by novices, we will reuse the knowledge gained from known instruments.

All the approaches we presented agree on our ability to reuse prior instrumental knowledge. For Rabardel, the "human" part of instruments resides in utilization schemes. For Beaudouin-Lafon, polymorphism is a feature of the instrument enabling its use in different situations. According to Rabardel's concepts of assimilation and accommodation, schemes can be enriched or transformed to be used with various instruments and in new situations. This point will be crucial for the rest of this dissertation. It means that users can reuse prior knowledge, even if the situation, the instruments, the objects of interest and the goals change. People who are novices regarding a new interaction technique can take advantage of previous experiences. This observation comes as an explanation and a far more precise description of what was called naturalness or intuitiveness in the introduction of that dissertation. It takes into account our advanced abilities to use technical and psychological artifacts in our everyday activities by considering the processes of internalization, assimilation and accommodation.

Beaudouin-Lafon and Mackay have proposed some dimensions to characterize instruments in HCI: the activation mode, the degree of (visual) indirection, the degree of integration and the degree of conformance. To account for its conformance, the instrument can communicate with the user through its reactions and feedback. According to Beaudouin-Lafon, an interaction model must be validated and evaluated regarding its descriptive, evaluative and generative powers. If it succeeds in providing all three of these powers to the designer, it will allow him to explore larger design spaces and create more efficient and innovative interaction techniques.

2.5 LIMITS IN THE EVOLUTION OF INTERACTION DESIGN

2.5.1 *The Reluctance To Design Complex Systems*

The cult of immediacy in interaction design does not acknowledge the richness of the knowledge and skills we all have thanks to our complex embodied and mediated relationship with our environment. In many interaction techniques, there is no focus on engagement, and no possible evolution from novice to expert. Furthermore, most of the time, the resulting interaction still requires users to go through

a laborious learning process. In a recent article, Djajadiningrat et al. complain about the current obsession for ease-of-use, arguing that it often just shifts the complexity from the motor actions to cognitive aspects (Djajadiningrat et al., 2007). For example, as pointed out by Beaudouin-Lafon, breaking down the command vocabulary in elementary graphical widgets in WIMP interfaces clearly restricts the gestural input to simple physical actions (e.g. clicking buttons, selecting items in menus), but could lead to high cognitive load during the exploration and memorization of the structure of the interface (Beaudouin-Lafon, 2000).

Interaction techniques based on in-air gestures lead to a similar situation. They avoid the necessity to learn how to operate a physical artifact, but they require the user to learn how to deal with the underlying gesture recognizer and discover the available commands and associated gestures. WIMP interfaces were necessary for the democratization of computer systems, and gestural interaction is a very rich input method, the exploration of which is facilitated by the proliferation of new sensing devices. But in these two cases, the ease-of-use will still depend on the complexity of the task and the design of the interaction. As a result, some WIMP interfaces are not efficient, and some gestural interfaces can be cumbersome to use. This assumption meets Cassell's statement in her article on speech gestures: "I don't believe that everyday human users have any more experience with, or natural affinity for, a "gestural language" than they have with DOS commands"(Cassell, 1998).

We can easily find historical and social reasons why interaction designers are reluctant to introduce a complex relationship between humans and computers. First, the marketing of computer systems always aims to increase the number of users and the number of features. As noted by Norman, immediacy is favored since the introduction of the one-button mouse by Apple in the 1980s, in order to get more customers by reducing the complexity of the physical access to their computers (Norman, 2010a).

Blackwell defends the idea that visual appeal of the visual interface of the Macintosh did not do justice to the rich relationship which should be provided by true metaphorical interfaces: "Designers were accused of making interfaces that are "cute," engaging the user's visual attention in a frivolous manner that is wasteful of mental resources" (Blackwell, 2006). Today, both these marketing-driven approaches have turned out to limit learning and transition to an expert state. The number of features is also an aspect that could be numerically measured and then easily used in advertisement. As Norman said, "when it comes to people, not everything we believe to be important can yet be measured. On the other hand, much that we know as unimportant is easy to measure" (Norman, 2010a). In fact, providing an apparent and measurable richness based on constantly increasing

lists of features does not seem to be the best way to take advantage of the complexity humans are able to tackle in their interaction with the world, and could even be confusing for both choosing devices and using them (see Beaudouin-Lafon, 1997).

2.5.2 *Investigating Accessible Complexity*

We argue that we can take advantage of complex artifacts that have already been mastered by common users. Their expertise in experiencing the world in an embodied manner, and in mediating their activities must be taken into account. In the previous sections, we have shed some light on important aspects of our embodied and mediated interactions with our environment which cognitive theories seems to fail at uncovering. The approaches we presented allow the description of skill acquisition and reuse.

Some interaction models, such as Instrumental Interaction, place our abilities in mediating our activities at the center of the design process. Nevertheless, even though they allow for the description and analysis of physical and digital instruments, they do not focus on taking advantage of psychological artifacts users have built. As Tangible Interaction (Wellner, 1991) makes use of the human expertise in manipulating physical objects, we propose to design interaction techniques which *reify* psychological artifacts users are used to deal with, as described in Activity Theory. In this manner, interactive systems can benefit from prior knowledge coming from activities where common users have coped with the complexity of expressiveness, and will not require intense learning. We do not waste their previous experiences, so that new interaction techniques can be more easily "*ready-to-interaction*" rather than being "*present-to-interaction*". Furthermore, novices have internalized some parts of expert activities through the active process of perception, and thanks to the intersubjectivity of human activities. Given the two processes of assimilation and accommodation, they are able to reuse the schemes they have built when perceiving the activities of others, or when communicating about them.

3

TAKING ADVANTAGE OF THE EXPERTISE OF "NOVICES"

> *"What are the people good at ? Language and art, music and poetry. Creativity. Invention. Changing, varying the manner of doing a task. Adapting to changing circumstances. Inventing new tools. Thinking of the problem in the first place. Seeing. Moving. Hearing, touching, smelling, feeling. Every one of these things is hard for a machine. Enjoying life. Perceiving the world. Exploiting taste (food), smell (flowers), feelings (amusement parks), body motions (sports). Aesthetics. Emotions such as joy and love and hope and excitement. And humor and wonder. But these are not the humans that technology see." (Don Norman, 1993, "Things that make us smart: Defending human attributes in the age of the machine", Basic Books, p223)*

In this chapter, we present studies in psychology, cognitive science and neuropsychology supporting the approaches to expertise proposed by phenomenology and Activity Theory, and validating the idea that non-experts can extract a sizeable knowledge from embodied perception of expert activities. During this perceptive process, they use their natural abilities in sense-making and social interaction.

Various processes allow us to naturally make sense of very complex activities that we perceive. Infants learn to speak by first building a more and more refined perception of the vocal sounds they hear. They progressively interpret speech phonemes, intonations, rhythm and body movements to understand words and sentences. Then they become able to imitate speech, link language to their subjective experience of the world, and use it to communicate (Rochat and Passos-Ferreira, 2009). In the rest of this dissertation, we will focus on expert activities in which non-experts naturally engage and for which they spontaneously use particular psychological artifacts, much like infants engage in verbal communication and develop language-related skills. We promote the idea that such engagement provides users with an implicit but rich knowledge that can be easily reused by designers to make expressive interactive systems "intuitive".

Artistic activities meet this condition since most people develop advanced abilities to appreciate art without *deliberate practice*, i.e. "prac-

tice that focuses on tasks beyond your current level of competence and comfort" (Ericsson et al., 2007). For instance, non-musicians perceive the tonal and rhythmic relationships between musical events (Palmer and Jungers, 2003). Artists and amateurs share a common cultural knowledge on which artistic activities are grounded, and which is constantly refined according to current aesthetics. Expressing oneself through art requires a lot of deliberate practice, but the activity of an audience appreciating it is not negligible either, even if they are not conscious of their skills.

In the following sections, we will describe what types of skills and artifacts that the musicians and the audience use in their respective activities of playing and appreciating music. Music has been recognized many times as a "cultural need", i.e. an essential part of our culture present in our everyday lives. non-musicians have a rich *tacit knowledge* of it that they gain by listening songs, seeing music videos or attending live performances. Our focus on music in this dissertation can be extended to other activities in which non-experts are engaged to a similar extent. For example, sportsmen perform complex moves to achieve specific goals according to the rules, which are precisely understood by most people (Schirato, 2007). In fact, sport fans perceive sportsmen' actions and intentions in such an embodied manner that their brain functions to process sport-related language are improved (Beilock et al., 2008). The study of dance leads to similar observations: while dancers express themselves through complex body movements, people in the audience are able to feel the effort and message of the dancer, as well as appreciate the quality of the performance (Dyson, 2009; Jola et al., 2012). To support such investigations, we will also provide general considerations on how human beings learn through perception and imitation.

3.1 THE STUDY OF EXPERTISE

3.1.1 *Expert Skills*

The study of expertise in human activities started in the 1950s and became one of the key topics of cognitive psychology in the following decades. In a recent book, Swanson and Holton provide a framework to help employees develop expert skills. They define expertise as a "displayed behavior within a specialized domain and / or related domain in the form of consistently demonstrated actions of an individual that are both optimally efficient in their execution and effective in their results" (Swanson and Holton, 2001, p241). In this section, we give an overview of the literature on expert abilities and processes, before studying how knowledge can be gained though observation.

All authors agree on the fact that a substantial amount of *deliberate practice* is unavoidable to become an expert. Deliberate practice helps

develop expertise by focusing on weaknesses and helps the novice learn how to fix them thanks to adapted exercises. Anders Ericsson, one of the spearheads of the study of expertise in psychology, argues that approximately 10000 hours of deliberate practice are necessary to reach world-class status in an expert activity. He states that expert performance "is the product of years of deliberate practice and coaching, not of any innate talent or skill" (Ericsson et al., 2007). He argues that objective measurements like IQ tests do not allow one to distinguish experts from novices. However, he presents teaching and coaching as central processes in expertise development. Furthermore, informal practice provides poor results if subjects do not know how to improve their actual abilities, and they even need different kinds of teachers at different stages of expertise to push their limits and continue developing skills (ibid.).

A description of expert abilities has been reported in the previous chapter in Dreyfus' interpretation of Merleau-Ponty's work: experts are able to instantaneously perceive and analyze the key aspects of a situation and know how to act to fulfill their goals. Such an efficiency requires at least advanced perceptual skills, motor capacities, and appropriately adapted artifacts. In the seminal "Cambridge Handbook of Expertise and Expert Performance" edited by Ericsson et al., authors present various approaches studying the aspects, structure and acquisition of expertise, and provide concrete examples from some professional fields, games, sports and artistic activities (Ericsson et al., 2006). They propose three main points of view to describe expertise:

- as the extrapolation of everyday skill to extended experience;
- as qualitatively different organization of knowledge;
- as reliably superior performance on representative tasks.

The main goal of this book, presented by Ericsson in the first chapter is to "understand how experts became that way so that others can learn to become more skilled and knowledgeable" (Ericsson et al., 2007). They propose a description of the universal skills developed by experts from all their practical studies:

- Qualitative analysis of problems;
- Detection and recognition of fine features;
- Planning of efficient strategies;
- Generation of the best solution, faster than novices;
- Monitoring of errors and corrections;
- Going through the previous points with a reduced cognitive effort.

Another important educational psychology book was published in 2000 by the U.S.A. Committee on Developments in the Science of

Learning (2000). The authors also support the idea of the structured perception and action of experts. From various studies on chess players, electronics technicians, physicians and other experts they identify four core expert abilities:

- Knowledge organization in order to perceive a situation in a hierarchical manner by identifying "chunks" of information;
- The use of context dependent knowledge retrieved with little attentional effort;
- The prediction of the available actions and the resulting evolution of the situation;
- The ability to adapt knowledge to new situations, and evaluate its limits.

The expert skills reported in these two books not only validates expert perception and action described by Merleau-Ponty, but also the definition of internalization proposed by Vytgotsky and the processes of assimilation and accommodation defined by Piaget. The cognitive load of perception and action is reduced when these processes are automatized as operations, and experts are able to simulate the activities they have internalized. They can also reuse utilization schemes with other artifacts and in other situations. It is worth mentioning that, in many activities, experts also develop particular motor abilities in addition to cognitive skills (Ericsson et al., 2006). They perform physical gestures with more spatial and temporal precision and they do not have to focus on every single movement they perform. Expert expressiveness is based on the implicit vocabulary provided by these cognitive and motor skills, and their combination thanks to the "grammatical" rules of the activity.

3.1.2 *Communities Of Practice*

In his practical work aiming to help organizations manage their knowledge, Etienne Wenger underlines the importance of the social dimension within expert fields. He presents the concept of "communities of practice", defined as social groups within which the meaning shared by experts is formed, negotiated, developed, and communicated (see Wenger, 1999). An expert must develop the skills of the community, and exercise them as one of its members. In many common human activities, a similar situation is faced: meaning exists not for a single individual but for a community of practice. It is the case for language, work organization and many other social activities including music. We share intersubjective understandings, expectations and significances.

Artifacts used by such communities, whether they are practical or psychological, reflect this common knowledge. They allow subjects

to externalize parts of the activity and transmit them to other people. As stated by Kirsh, they "serve as a repository of knowledge" (Kirsh, 2009). In his article on the evolution of artifacts, Kirsh illustrates this statement with the study of the flute: the flute has been made for the particular activity of playing music, and novice musicians learning how to play the flute have their experiences shaped by the structural and functional properties of this instrument. Therefore people, artifacts and activities co-adapt when the practices and goals of a community of practice are modified. A similar idea is defended by Norman in his book *Things that make us smart*: "We organize things in a way that could provide us information later on. Structure the world to improve and facilitate the later interaction we will have with it" (Norman, 1993).

We neither go into detail about distinguishing experts from novices, nor about the general description of expert artifacts, since music expertise, musical concepts and musical instruments have received particular attention in many fields, and will be presented in the following sections. In summary, these approaches promote the idea that advanced skills are gained thanks to the knowledge encapsulated in expert artifacts, efficient learning methods, and deliberate practice. The observations presented in this section are consistent with embodiment and Activity Theory. However, several researchers in psychology, cognitive science and neurology focus on how skills can be developed via the complex sense-making mechanisms that constitute social interaction.

3.2 BUILDING ADVANCED TACIT KNOWLEDGE THROUGH PERCEPTION

In this section, we present studies from various fields supporting the idea that people gain significant knowledge from the activities they observe. They are able to grasp the intentions of the people they observe, understand it from their point of view, build psychological artifacts to structure their perception, and even partly reproduce the physical actions they perceive. This argument was supported to some extent by the discovery of mirror neurons in 2004 (Rizzolatti and Craighero, 2004), which are active during both perception and action, and therefore may help the observer internalize the actions of the performer both psychologically and physically.

3.2.1 *Tacit Knowledge*

In the late 1960s, Michael Polanyi has studied knowledge and skills that can neither be represented in propositional forms nor made directly conscious by subjects, referring to it as *tacit knowledge* or *implicit*

knowledge. This idea is summarized in his famous assumption that "we know more than we can tell" (see Polanyi, 1966). The tacit knowledge we gain from unsupervised experiences is often situated and difficult to explain to others. Chomsky also argued for the central role of tacit knowledge in his study of language syntax (see Chomsky, 1965). For him, one can learn one's mother tongue during the elaboration of the process of communication, without being taught the technical aspects of grammatical rules. But in the end, these rules are implicitly known since the language as a whole is grounded on them. Since such knowledge can be rich albeit built without ever going through arduous sessions of deliberate practice, we can reuse it to design expressive and accessible interaction techniques. In fact, implicit knowledge is central to our experience as social beings.

3.2.2 *Social Interaction*

Maturana and Varela insist that "the phenomenon of communication depends not on what is transmitted but on what happens to the person who received it" (see Maturana and Varela, 1992, p196). Artistic activities provide one of the richest illustrations of this assumption. For example, when observing a painting, one receives as input far more than colors and shapes. Even if the painting is not figurative but abstract, i.e. if the meaning conveyed by the art piece is not explained and its understanding is not guided, amateurs are still able to appreciate it deeply and perceive complex intentions. During social interactions, a space for intersubjective meaning—mostly implicit—emerges between people, even if they are not able to provide a precise formulation of what they experienced. In his book on language, Herbert Clark introduces the notion of "common ground" to describe this interpersonal meaning necessary for interpreting and understanding utterances (Clark, 1996). This common ground is highly dependent on the subjective experience of each of the people who are interacting, but it nevertheless includes all the intersubjective and tacit knowledge, which is required for communication. We can even say that in societies, the perceptions we have of the world and the meaning we assign to phenomena mainly emerge during social processes such as observation, teaching or communication.

These intersubjective experiences have been deeply studied in the recent years by authors defending the *enactive approach* to social interaction such as Hanne De Jaegher, Ezequiel Di Paolo or Thomas Fuchs. They study social interaction as a coupling between individuals where intersubjective meaning emerges, rather than referring to personal internal representations as it is defended by information theory. For them, social cognition is "the interactive, interpersonal generation and transformation of meanings" (De Jaegher, 2010). De Jaegher et al. describe human beings as systems that are constitutively au-

tonomous and preserve their proper organization while being able to adapt to their environment and to other subjects (De Jaegher and Froese, 2009). The fundamental social nature of human cognition based on these two processes allows humans to elaborate common meaning about the world through *participatory sense-making*. The proficiency of humans at social interaction provides this process with the richness and directness we all experience during conversations where meaning is constantly negotiated. We can even acquire new skills and understandings about our own experiences during social interactions.

For Fuchs et al., participatory sense-making arises thanks to the coordination of the body movements, utterances, gestures, gazes, etc. (Fuchs and De Jaegher, 2009). The authors introduce the notion of *mutual incorporation* as a process in which the body and experiences of the subjects expand and, "in a certain way, incorporate the perceived body of the other" (ibid.). Mutual incorporation is the phenomenal basis for social understanding that includes a proprioceptive component because of our embodied nature. The authors illustrate this physical aspect through the analysis of the movements of skilled tennis players. The movement of the body of each of the players and the way they will hit the ball depend not only on the trajectory of the ball, but also on the perception they have of the body of their opponent. If the opponent decides to move towards the net, the player can perceive this decision and try to send the ball to the back court. Therefore, his decisions depend on the embodied perception of the opponent's body.

Mutual incorporation is also at stake in artistic activities in which artists perform in front of an audience. During concerts, the audience experiences an intense perceptive process through which they receive the intentions of the performers, which in turn depends on the reaction of the audience. The resulting performance is thus shaped by the mutual incorporation of the musicians and the audience. We can even say that this two-way social interaction between performers and audiences is one of the grounds of musical activities. Fuchs et al. go further in the description of mutual incorporation, asserting that it could go from joint sense-making, to *guidance* or *fascination* where only the fascinated person incorporates the intentions of the person by whom he is fascinated. The former is thus guided in his perception and sense-making to an extreme extent.

3.2.3 *The Embodied Grounds Of Social Sense-making*

Gallagher et al. state that the intentions of other people, grasped through the perception of their embodied actions, are mirrored in our own capabilities for action (Gallagher and Hutto, 2006). Contrary

to cognitive approaches inspired by Information Theory, the enactive approach to social interaction allows the description of infants' sense-making. As stated by Gallagher, "they are able to see bodily movement as expressive of emotion, and as goal-directed intentional movement" (ibid.). The situated perception of others and the observation of their actions and the ways they are bodily expressed is sufficient for grasping their purposeful intentions. Thanks to these perceptive processes, infants build a non-conceptual, action-based and pre-conscious understanding of others by the end of the first year of life (ibid.). At this age, infants do not try to conceptualize the desires or beliefs of people they observe, they just directly perceive intentions. This innate ability of humans is called "primary intersubjectivity" by Trevarthen (Trevarthen, 1979).

If intentions can be grasped, the question is: can this embodied perceptive process allow us to reproduce actions? Infants actually use another innate ability for social interaction, which is imitation. They naturally and unconsciously mimic the actions of the adults they observe, as the first attempt to be recognized as members of the human community. In his book on imitation, Andrew Meltzoff reports several studies on infants (Meltzoff, 2002). Newborn infants are able to imitate "lip and tongue protrusion, mouth-opening, hand gestures, hand movements, cheek and brown motions, eye-blinking, and components of emotional expression". They spontaneously turn new visual information into a motor action, without going through any intellectual process. In fact, innate imitation has been recognized as the basic mechanism by which infants develop empathy and the capability to think and speak.

The concepts of participatory sense-making and the studies on innate imitation are supported by the discovery of mirror neurons in the late 1990s. Giacomo Rizzolatti and his colleagues discovered neurons that are active when an individual performs an action, as well as when he observes a similar action done by another individual (Rizzolatti and Craighero, 2004). This contribution to neurosciences is often described as direct evidence of the common neural structure of action and perception. In fact, the activity of these neurons during perception may illustrate a sensory-motor coupling involved in embodied understanding of the others' intentions. The activation of mirror neurons is implicit, automatic and requires neither any conscious cognitive effort nor conceptualization of another's actions. It has been observed in various situations. Visuo-motor neurons play an important part in imitation. Somatosensory mirror neurons, which match observed and felt touch, are involved in understanding the effect of tactile stimulation on others (Blakemore et al., 2005). The activation of mirror neurons always remains below a certain threshold, so that no conscious sensation of touch or movement is experienced. Further studies reported the role of mirror neurons in understanding the

expressive aspects of actions or even emotions (Calvo-Merino et al., 2005). In summary, mirror neurons allow subjects to internally simulate the gestures of others, participate to imitation, and may play a crucial role in embodied social understanding and imitation.

In companies and schools, the practice of work shadowing takes advantage of this natural ability to extract embodied knowledge from observation. Work shadowing means that "one person (the shadow), visits another (the host) to experience their work by observing them for an agreed period of time" (see of Education and Training, 2009). It can be used in a work team to get a deeper understanding of the roles and functions of other teams.

In her book on intersubjectivity, Fransesca Morganti describes this research field as a triangulation between cognitive sciences, social sciences and neurosciences aiming to understand the central social dimension of sense-making in humans (Morganti, 2008). The studies in psychology and neurology presented in this section define the enactive knowledge acquired in social interaction as the most natural kind of knowledge. It is based on our most innate abilities in sense-making and it is only grounded on perceptive processes that do not require deliberate practice. People who are novices in an activity are still experts in social interaction, participatory sense-making, extraction of intentions through observation, and imitation. Communicative activities, including artistic expression, illustrate the use of these skills, where the audience's perception and the artists' expression are based on the same psychological artifacts, e.g. syntactic rules, aesthetics or expressive features. The use of common psychological artifacts such as symbols in participatory sense-making is called "secondary intersubjectivity" (Rochat and Passos-Ferreira, 2009). Before describing what kind of music-related skills have been measured on listeners and observers, we will provide an overview of the cognitive and motor abilities of expert musicians.

3.3 MOTOR, COGNITIVE AND EXPRESSIVE ASPECTS OF MUSICAL EXPERTISE

Playing music is one of the most complex human activities. As stated by Eric Clarke in the late 1980s in the seminal book on generative processes in music edited by John Sloboda, "playing music is an activity that is comparable in cognitive complexity to speaking a language, and comparable in its demands on motor control to playing a sport like tennis" (Clarke, 1988, p1). Music instruments are surely among the most advanced artifacts used for expression in human history, but the psychological artifacts of music are also particularly rich. In this section, we describe the expert skills developed by musicians.

3.3.1 *The Structure Of Music Knowledge*

Western tonal music is based on large vocabularies of *notes* made of *durations* and *pitches* or *tones*, that can be vertically combined in *chords* following the rules of *harmony*, and played in articulated sequences such as *melodies* and *chord progressions*. *Tonality* defines hierarchical relationships of similitude and difference among *pitches*. The *key* of a music piece defines its tonal hierarchical structure, so that the pitches are considered *consonant* or *dissonant* regarding the *tonic* of the key, which is its most central component. In this manner, tonality defines a hierarchy of relative importance among pitches in a key. Temporal aspects of music, regrouped in the *metric structure*, have a fundamental importance too. Metric structure includes *tempo* defining the global pace at which musical events occur, *meter*, i.e. the periodic and regular accentuation of musical events, and *rhythmic aspects*, i.e. the relative durations and grouping of musical events and interleaving breaks. *Music genres* bring additional syntactic and semantic rules restricting the space to explore inside harmony, tonality and metric structure. For example, only a few rhythms and keys are to be used in traditional jazz, while free jazz improvisation aims to get rid of constraints of that kind.

An explicit knowledge of how to practically use these rules leads musicians to a higher level of expressiveness. Sloboda reports an experiment where musical achievement reveals to be highly related to the amount of formal practice, while having weaker relationships with informal playing (Sloboda et al., 1996). He argues that approximately 10,000 hours of deliberate practice are necessary to become an expert player, not only to develop enough *technique* for the accurate rendition of the music, but also to play in an expressive manner. However, some good musicians have declared having only an implicit knowledge of it gained through extensive listening and informal practice of a music genre without any formal teaching (e.g. Django Reinhart, John Coltrane, or even Arnold Schoenberg who radically changed the rules of Western tonal music).

3.3.2 *Motor Skills*

Playing music first requires advanced motor skills. Alan Watson describes the physical aspects of instrument playing as being "close to the bounds of what is physically possible" (Watson, 2006). Professional pianists reach extreme levels of bi-manual abilities like hand separation or synchronization. They not only exhibit more independence of finger movement than amateurs, but also a better control of the duration and force of hand movements (ibid.). Skilled pianists have the potential to produce movements at rates that exceed visual reaction times (Thompson et al., 2006). Jazz guitar players switch between complex finger configurations at a very high rate.

Many traditional forms of music combine simple rhythmic structures played in parallel to build up higher level aggregated rhythms, called "polyrhythms" (Arom et al., 1991). Highly trained musicians are able to create and play an incredible amount of rhythmic variations. With practice, the motor sequences required to reach musical goals are automatized as operations, so that the performer can concentrate on expression (Gruson, 1988).

We all experience the same level of operationalization when speaking in our mother tongue without concentrating on the complex motor aspects of vocal sound production. Musical instruments progressively become functional organs, and expert performers can reuse the utilization schemes they have learned in other situations: when playing a new musical sequence, they will perform it more accurately if it shares motor or structural aspects with sequences they have learned before (Palmer and Meyer, 2000). Expert musicians, as in most other fields of expertise, can even rehearse mentally (Watson, 2006), which means that the motor abilities they use to play their instrument and the corresponding perceptive feedback have been fully internalized in the terms of Activity Theory. During this virtual practice, both the perceptive and motor regions of the brain that are normally active while playing are activated.

Specific learning methods allow one to deal with these complex motor actions in a progressive manner. For instance, piano teachers encourage students to first play chords as arpeggios, pressing the keys one after the other, to experience a step-by-step understanding of a chord. With that structural knowledge, they are then able to build more complex chords as alterations or diminutions of basic ones. Other approaches to the acquisition of motor skills, e.g. Dalcroze's Eurhythmics (Jaques-Dalcroze et al., 1930), support the idea that elementary motor actions are easy to learn in a separate manner, before being combined in more complex actions. The learning method proposed in chapter 5 to teach chording gestures is in fact inspired by such approaches.

3.3.3 *Cognitive Skills*

In Sloboda's book, Linda Grusson presents perceptive and motor aspects of musical expertise which correspond to the enactive approach, even if this perspective is not established by the author: expert musicians are able to process larger and more complex units of meaningful information, and are able to organize perception and action in a hierarchical manner (Gruson, 1988). For example, professional pianists read chunks of up to 5 notes in advance when sight-reading, i.e. when playing a piece from a score they have never seen. They read groups of notes that are tonally consistent, since they are

able to immediately perceive such tonal relationship. On the contrary, non-professionals sight-read analytically and note-by-note (ibid.).

Caroline Palmer studied several cognitive skills of expert musicians. She presented studies showing better anticipation, planning, accuracy, sensitivity to musical structures, expressiveness, error detection and correction in expert musicians (see Palmer and Drake, 1997; Palmer and Jungers, 2003). For example, intermediate pianists anticipate motor events and they can start the movement of a finger up to 3 notes before it strikes the piano key.

3.3.4 *Musical Expression*

One of the skills that are shared by all musicians is expressiveness. Expressive aspects of music regroup the composer's message embedded in musical structure, and the expressive intentions of the performer's interpretation. Even with the same score of a classical piece, skilled performers add variations in timing, pitch, timbre or dynamics to the musical structure (see Palmer, 1997). For example, onsets of the melodic voice can precede other voices onsets in notated simultaneities, and a slowing in tempo often occurs at phrase boundaries. The metrical structure is often emphasized by lengthening the durations of the notes corresponding to metrical accents, and playing them with smoother transitions. An increase in tempo and loudness will be used to express happy or angry emotions, while these two features will be decreased to express sad emotions (ibid.).

As musical expression mostly lies on structural features, music is often studied in comparison to language. Notes are hierarchically grouped according to syntactic rules. Chords, chord progressions and keys, have the same role as words, clauses and sentences in language (Patel, 2003). In fact, the succession of chords made of notes having different relations with the key leads to a succession of structural tensions and resolutions that are syntactically comparable to the organization of clauses in language.

Expert musicians also internalize all these structural, expressive and syntactic rules. In a study reported by Davidson et al. (1988), first year and third year music students are asked to compose a tune. The former are unable to construct a strategy and cannot explain their goals until they have finished solving the problem. They randomly hit keys and need to hear the resulting sounds to decide if they want to keep these as part of the final composition or not. On the contrary, third-year students were able to define goals and methods from the beginning of the test. The internalization of the composition task allows them to know what outcomes they can obtain both structurally and acoustically. From the point of view of Activity Theory, we can say that the psychological artifacts used for composition have started becoming functional organs for these advanced students. Practice

can then be focused towards this internalization: students that learn tonal, harmonic and syntactic aspects of music structures will just need the chord grid of a new music piece to be able to play along with it (Bødker and Klokmose, 2011).

Expressing oneself with the language of music obviously demands significant deliberate practice, oriented towards the development of advanced motor and cognitive skills in order to use music instruments and language properly. Expert musicians are able to deal simultaneously with complex movements, intricate rules, music genres and reach a level at which planning, action, perception and correction are instantaneous. But since music is a means of expression and communication, non-musicians also have an implicit knowledge in order to make sense of it. Listeners and observers can use their embodied perceptive abilities in two situations to enact music knowledge: when listening to music or seeing live performances.

3.4 MUSIC PERCEPTION: IMPLICIT INTERSUBJECTIVE KNOWLEDGE AND MOTOR SKILLS

Listeners often show strong musical tastes and appreciate the differences between various performances of a song. Reactions of the audience are as rich when attending concerts as when attending plays, even if they are not guided by language and do not see an explicit illustration of emotions. non-musicians are able to differentiate lip synchronization and imitated instrument playing from real live performances, and to appreciate the expressive features added to musical structure. These abilities attest to their advanced perception and understanding of music, even though they do not practice it formally.

As an expressive activity, music is based on a constant negotiation between performers' expression and listeners' understanding. non-musicians indirectly shape music production with their tastes, and gain cognitive and even motor skills from their exposure to musical structure and expression. This knowledge, whilst largely implicit, is considerable and allows non-musicians to make sense of musical structure without being aware of the explicit rules of Western tonal music (see Bigand and Poulin-Charronnat, 2006).

3.4.1 *Human Musicality*

First, a common ground for sense-making in music rests on the embodied nature of humans (Pelinski, 2005). Performers and listeners share common neural structures, perceptive abilities and motor possibilities. All humans have a natural perception of rhythm, thanks to periodic mechanisms that are involved in the internal functioning of our organism (heart beat, sleep cycles, breathing, etc.) (Glass, 2001).

Several researchers argue that the motor system provided by our bodies responds to "internal clocks" defining common spontaneous tempo for movement (Trevarthen, 2000). In fact, periodic actions like chewing or walking are known to have universally preferred rates (MacDougall and Moore, 2005), and all listeners show preferences for music pieces with tempi around 300-900 ms per beat (Palmer and Jungers, 2003). Some approaches to music learning, like Dalcroze Eurythmics (Jaques-Dalcroze et al., 1930), are actually grounded on the training of this innate bodily sense of rhythm.

Our natural abilities to perceive the elementary features of music and make sense of these play an important role in human development. Several embodied reactions to pitch, rhythm and structure of speech have been observed in infants before they learn the narrative aspects of language. These reactions are not only described as the grounds of sense-making, but also of empathy, coordination and communication (Gill, 2007; Malloch, 2005). This natural common origin of social understanding and communication, and likewise with music production and appreciation is termed *human musicality* by Malloch and Trevarthen (Malloch and Trevarthen, 2009).

Musical structure and its perception share the kinematics of movement as a common origin (Palmer, 1997). Similitudes between musical motion and physical motion are often observed. For example, performers reduce the tempo near phrase boundaries at a rate similar to slowing down from a run to a walk (Palmer and Jungers, 2003). In various articles, Wayne D. Bowman describes music listening as a mode of musical engagement, and stresses the centrality of embodiment to musical experience, for both the performer and the listener (see Bowman, 2002; Bowman and Powell, 2007).

3.4.2 *Making Sense Of Music*

Beyond these natural predispositions, advanced perceptive skills are progressively acquired through interaction with a large number of music pieces respecting the rules of western tonal music. The regularity with which tones, chords and keys occur provides enough information for listeners to identify the key of a music piece, its tonic and perceive similarities and distances between tones (see Palmer and Jungers, 2003; Patel, 2003; Tillmann et al., 2001). They are thus able to experience the tensions and resolutions bounding melodic motifs or chord progressions. Listeners also induce meter by perceiving accents on tones that are played with dynamical or temporal emphasis. Small accelerations and decelerations in tempo, together with the hierarchical structures of tones, allow them to perceive relations between groups of musical events.

Thanks to the constraints of Western tonal music, which reduce the number of elements that are to occur, these sophisticated skills

are actually developed early in life. Around six months old, infants show preferences for appropriately structured music, and recognize melodies even if their starting pitches have been shifted (Palmer and Jungers, 2003). At age 5, children have a structured perception of rhythm, albeit contextual. By the age of 8 they acquire a better understanding of isolated rhythmic patterns, and become able to perceive hierarchical relationships among tones (ibid.). This knowledge acquired through simple perception of structured auditory phenomena influence future listening. non-musicians show strong expectations in harmony, tonality and rhythm. If music pieces do not correspond to these expectancies, listeners will tend to consider them as deficient (Palmer, 1997). Furthermore, violations in musical structure have been described as leading to the same neural reaction as language violations (Patel, 2003).

In addition to formal structure, listeners also integrate expressive variations to their perception. Palmer reports studies where expressive cues emphasize listeners' melodic expectancies and their perception of tensions and resolutions (Palmer, 1997). Performers often exploit that perception of variations as additional information about the structure (Sundberg, 1988), as actors would add variations in intonation or articulation to the original punctuation written by the author of a play.

Bigand et al. report several studies showing that listeners with sufficient exposure can be considered as "musically experienced listeners" since they are able to respond to music in a very sophisticated way (Bigand and Poulin-Charronnat, 2006). In fact, our exposure to auditory musical stimuli is large. We all listen to music everyday on TV, radio, computers, music players, etc.

The same authors critique previous experimental protocols that have led researchers to establish that musicians have a better perception of music than non-musicians, and propose to avoid experimental tasks for which musicians have been explicitly trained in conservatories. The authors show that musicians are not better than non-musicians in categorizing music excerpts regarding the emotions induced. Both perceive tensions and resolution in a similar way. Furthermore, non-musicians do not need more contextual information than musicians to take the current tonal context into account and anticipate events that are consistent with the key. This ability to anticipate events can be considered as an expert skill, as described in the previous sections. On the whole, musicians and non-musicians were shown to perform similarly in cognitive and emotional tasks measuring the quality of the perception of musical structure (Bigand and Poulin-Charronnat, 2006).

3.4.3 *Acquiring Motor Skills When Listening To Music*

Non-musicians are not only able to make sense of music, but they also develop music-related motor skills from their embodied perception of it. From childhood, humans are used to sing or whistle, tap their feet, clap their hands, snap their fingers or move in synchrony with music. Around six months old, infants' babbling in response to music often preserves the melodic contour (Palmer and Jungers, 2003). Three year old children are able to repeat short musical phrases accurately, and spontaneously produce novel musical phrases in song (ibid.). Moreover, adults are able to tap complex rhythms along with music (ibid.). Lahav et al. report that simple listening makes non-musicians able to learn how to play short and basic musical sequences on the piano, and even improves their motor performance on other sequences (Lahav et al., 2005). While abilities of non-musicians to develop motor skills from auditory perception depend on their personal action capabilities and prior motor experiences, Rodger et al. report studies indicating that such abilities have greater dependency on our common physiological nature (Rodger et al., 2007).

These observations seem to illustrate the activity of mirror neurons when listening to music, even if a two-way relationship between the auditory and motor areas of the brain has only been measured in experts (Watson, 2006). However, in an article defending dance and music as integral to being human, Stephen Malloch present studies showing that mirror neurons are multimodal in monkeys. Audiovisual mirror neurons show activity whether the perceived actions are heard or seen (Malloch, 2005). Rodger et al. argues that the activation of mirror neurons also depends on previous motor experiences: more mirror neurons will be activated in non-musicians since they will simulate various motor actions that they could imagine as corresponding to what they hear (Rodger et al., 2007). Even if this simulation will be less precise than for musicians, they are still able to reproduce the kinematics of the global musical movement (e.g., dancing, illustrating perceived loudness or changes in pitch) (ibid.).

3.4.4 *Visual Perception Of Musical Expression*

Leman describes auditory stimuli as sufficient to grasp the intentions of the performer, since music directly appears to them as an intentional organism including an acting subject (Leman, 2007). But additional clues are reachable by observing musicians during performances. People in an audience are engaged in a process of mutual adaptive behavioral resonance with the musicians on stage, called *entrainment* (ibid.). François Delalande distinguishes between two types of physical gestures performed by the musician (Delalande, 1988):

– effective gestures, providing physical energy to the instrument, so as to produce sound with it;
– accompanying gestures, which are movements of various other parts of the body.

Each of them convey specific information participating in the visual communication with the audience: effective gesture illustrates musical structure, while accompanying gestures, together with facial expressions, provide clues as to the effort of the performer and his own emotions during the performance.

Thompson et al. study how visual aspects of musical performances contribute to the communication between performers and listeners (Thompson et al., 2005). They found that visual aspects of performance such as facial expressions, body movements, and hand gestures influence the perception of the musical structure. Visual information makes listeners focus on temporal events, helps them clarify the pitch-related structure of music, highlights affective interpretation, guides emotional responses and even makes voluntary dissonance acceptable (ibid.). The authors also critique the fact that the relation we have with today's popular music is loosing these critical visual aspects supporting understanding, since more importance is now given to artificial visual styles having arbitrary relations with the musical message. Famous performers no longer show embodied relations with acoustic instruments and musical structure (ibid.). Dahl et al. also showed that the body movements of performers are sufficient to communicate emotions such as happiness, sadness or anger, even with no auditory stimulus (Dahl and Friberg, 2007).

In summary, these studies suggest that non-musicians have an advanced perception of the hierarchical structure and grammar of music including rhythm, tonal relationships, tensions and resolutions, and even expressive variations added by performers. non-musicians are able to grasp the intentions of musicians, i.e. primary intersubjectivity. Implicit musical knowledge of non-musicians as well as expressive content written by composers and interpreted by performers present the same structural and syntactic rules constituting the psychological artifacts of music, i.e. secondary intersubjectivity. Thanks to their embodied experience of music, non-musicians are able to move along with it and illustrate its structure with gestures.

3.5 EXPRESSIVE INTERACTIVE SYSTEMS USING NOVICES SKILLS AND KNOWLEDGE

3.5.1 *Non-experts' Knowledge Of Expert Activities*

Novice users are not incompetent. As human beings, they have acquired expert skills from embodied and social experiences, and

use them efficiently to cope with the complexity of their lives. Humans not only develop complex motor skills such as the ability to walk or run, but also complex perceptive, cognitive and social skills. In light of the philosophical, psychological and neurological studies presented in the previous sections, we identify four complementary kinds of natural expert abilities in humans:

- Making sense of their embodied interaction with their environment (phenomenology);
- Using technical and psychological artifacts to mediate their activities (Activity Theory);
- Imitating others' actions and grasping their intentions from embodied perception of social phenomena (neurology of social interactions);
- Perceiving, organizing, creating, negotiating and communicating intersubjective meaning (intersubjectivity).

Each of these innate skills allows us to gain knowledge. Embodied perception and understanding of our environment make us familiar with natural phenomena such as gravity. Using artifacts sculpts our activities and makes us learn the cultural knowledge embedded in their structural and functional properties. The last two skills are used during social interaction to understand others' actions, imitate them, build intersubjective meaning and share knowledge. These skills and the implicit knowledge we gain from our embodied, mediated and social activities can be reused in interactive systems to make them accessible.

Expert activities draw an even wider design space, since they are based on additional advanced skills and complex artifacts. They lead to a high level of expressiveness accessible thanks to deliberate practice. Even though interactive systems can be inspired by the richness of such activities and make use of existing efficient learning methods, the transition from novice to expert may take time. However, people who are novices regarding an expert activity can still be familiar with it as experts in social interaction. Apart from the expert activities that are necessary for living in society, e.g. verbal communication, human beings naturally engage in other expressive expert activities, such as artistic activities.

The studies on music cognition presented in this chapter attest to expert music-related skills and knowledge developed by non-musicians. Music is grounded on human familiarity with rhythm and pitch, coming from internal mechanisms and early communication, and non-musicians and musicians show more similarities than differences regarding their perceptive abilities. In fact, non-musicians can be seen as expert music listeners from three points of view:

- As having the physical and neuronal structures to perceive acoustic events in a structured manner;
- As experts in music listening who have implicitly internalized the language of music;
- As experts in social interaction and grasping intentions from perception, including listening and observation.

3.5.2 *Using Musical Knowledge In HCI*

Taking advantage of the implicit musical knowledge of non-musicians to design interactive systems can make them at the same time expressive, engaging and easily accessible. First, expert artifacts, such as musical structure, have been refined over time and are the most efficient tools for coping with complexity and for expression of human intentions. They allow deep internalization, stable practice since they have been standardized, and can be used as functional organs. Second, users will not be discouraged by the complexity of an expressive interaction technique if it is based on knowledge they have already internalized. Their skills can easily adapt to using such systems thanks to assimilation and accommodation. Third, music is considered an intrinsically rewarding experience that can contribute to happiness and well-being (Csikszentmihalyi, 1991). non-musicians enjoy their relation with music. Furthermore, they usually reach a high level of arousal when playing music video games. In Piaget's theory of constructivism, play improves understanding and internalization (Piaget, 1973). Research in HCI actually supports the idea that having fun improves learning and memorization (Bragdon et al., 2010). Fourth, depending on how much the actions required by the system are familiar to users, we can use learning methods inspired by music education to efficiently teach them how to use our interaction techniques.

Finally, non-musicians feel free with music, even if they do not have the same expressive power as musicians. Stephen Malloch describes music as more direct than language (Malloch, 2005) and Ian Cross states that music leads to fewer disagreements than verbal communication: "We can agree in the shared embodied space of music and dance, whereas we may disagree in the shared objective space of a verbal discussion because our version of reality differs from that of another" (Cross, 1999). From that point of view, interaction techniques based on music can be less frustrating than techniques based on language like speech recognition. We argue that making expressiveness accessible in this way leads to a rich interactive experience and does not restrict the design space to instantaneously usable interactions.

We also consider some additional requirements inspired by musical instruments and the language of music. In fact such artifacts allow novice as well as expert activities. non-musicians can easily play a few notes or simple chords along with music, and expert musicians can access a high level of expressiveness by mastering these tools. The problem of learning and transitioning from novice to expert have been described by Scarr et al. (Scarr et al., 2011). The authors state that the reluctance to use expressive interaction techniques comes from the switch to new techniques, the performance dip arising from that switch and the performance ceiling. We observe that non-musicians can easily perform actions corresponding to one aspect of musical structure. For example they can whistle a simple melody, tap complex rhythms along with music, or illustrate the kinematics of the global musical movement when dancing.

Figure 2: The Guitar Hero note chart and input device. While the note chart scrolls down, the non-dominant hand presses the fret button having the same color as the next note, and the dominant hand strums the strum bar when that note reaches the empty circles at the bottom of the screen. The visual feedforward and auditory feedback allows the player to perform complex bimanual multifinger rhythms with little practice (Activision Publishing)

Taking advantage of these motor skills is sufficient to define a large design space and create interaction techniques with which novice users will not fear a drop in performance since they will not start from scratch with the new technique. But when playing musical video games such as Guitar Hero[1] (see Fig. 2) which makes users go through a quick learning process, they need little practice to become able to perform more complex gestures like bimanual multifinger rhythms. Such gestures are actually very expressive and correspond to a high performance ceiling. Therefore, we must establish a distinction between initial accessibility that allows a novice to start

1. Guitar Hero website: www.guitarhero.com

using the system, from access to expertise which paves the way to the development of expert skills and internalization.

Scarr et al. also define design guidelines to improve the transition from novice to expert with expressive interaction techniques (Scarr et al., 2011). First, we must minimize the syntactic differences between novice and expert modes to favor users' engagement in skill acquisition. The first skills they acquire when using the interface must not be wasted when they become experts. Second, our interaction techniques must not change the semantic aspects of interaction by giving access to the same vocabulary of commands and actions as usual techniques. Third, high performance ceilings are favored by features such as spatial predictability, low display demands and flat command structures (ibid.). Furthermore, the studies we presented in phenomenology support that novices must be guided during the learning process and the features of the interface must be progressively presented. Research in HCI confirms that observation: use and decision-making must be guided in the beginning, and tasks that are already known must be accelerated (Wu, 2000).

In order to know when the expert level has been reached, Bannon and Bødker state that internalization can be quantified by observing whether users' actions have been automatized as operations, and whether they mostly focus on the actions they have to perform or on their results (Bannon and Bødker, 1991). Situations in which users focus on understanding the system rather than on their main activity are considered as "breakdowns". Bødker and Klokmose identify three types of breakdowns (Bødker and Klokmose, 2011). Breakdowns at the operational level occur when users focus on motor skills required to use the system. Breakdowns at the handling level occur when the functioning of the system has been misunderstood by the user, revealing improper training or inconsistencies in the design. Breakdowns at the motivational level are products of a mismatch between the goals of the user and the outcomes accessible with the system.

3.6 A FRAMEWORK FOR USABLE AND EXPRESSIVE INTERACTION INSTRUMENTS

We operationalize our approach by drawing upon Instrumental Interaction (Beaudouin-Lafon, 2000) and the Human Artifact Model (Bødker and Klokmose, 2011) and consider the user from the psychological point of view in light of the studies we presented in the previous sections. We distinguish between the appropriateness of an instrument to act on the objects of interest, which we call its *effectiveness*, and its *usability*, i.e. its adaptedness to users skills, knowledge schemes, utilization schemes and sense-making abilities. Regarding these two definitions, we restrain the study of *efficiency* to

the measure of the time spent to achieve the desired effect on objects of interest. Many studies in HCI concentrate on efficiency by comparing temporal measurements of tasks such as pointing a target or reaching an item in a menu. Within the Human Artifact Model (Bødker and Klokmose, 2011), the properties proposed by Instrumental Interaction to describe and compare instruments characterize the relationship between the operational and instrumental aspects of instruments, i.e. their *effectiveness*. We rather focus on the relationship between the subject's capabilities and the operational level of instruments, i.e. its *usability*. The instrumental and motivational aspects (Bødker and Klokmose, 2011) are not considered in the projects we present in the following chapters since our goal is to provide interaction techniques inspired by music practice as alternatives to existing techniques in given contexts of use.

We borrow the term *usability* from the ISO/IEC norm on Product Quality in Software Engineering (ISO/IEC, 2001). In this norm, the study of software *usability* implies the analysis of *understandability*, *learnability*, *operability* and *attractiveness* [2]. We propose a framework to study these four criteria together with *expressiveness* in interaction instruments, from the point of view of enaction and social psychology.

3.6.1 *Understandability*

An instrument is *understandable* if the user knows what actions it takes into account, understands its reaction, and can structure this knowledge. *Accountability* is necessary for an instrument to be *understandable*. First, *consistency* must be ensured. For example, an action performed several times in the same situation must lead to the same reaction of the instrument. Second, the functioning of the instrument can be shown to the user as an *abstraction* presenting salient aspects in order to ease *understanding*. The instrument must at least provide feedback to inform the user when an action has been taken into account.

Understandability will be maximized if instruments put the user in familiar situations, so that he can reuse existing schemes or easily adapt them through assimilation or accommodation. That is the aim of speech interaction, metaphors of common technical artifacts such as digital calendars, and metaphors of the physical behavior of our environment. This latter category is illustrated by projects such as the Bumptop system (Agarawala and Balakrishnan, 2006) or physics-based interaction for multitouch surfaces presented by Wilson et al. (2008). In the Bumptop desktop interface, files are represented as 3D objects having a mass and a volume (see Fig. 3(a)). Pen-based interaction techniques allow the user to organize them spatially by

2. In the reste of the study, when these terms are in italics, they refer to the definitions provided in this section

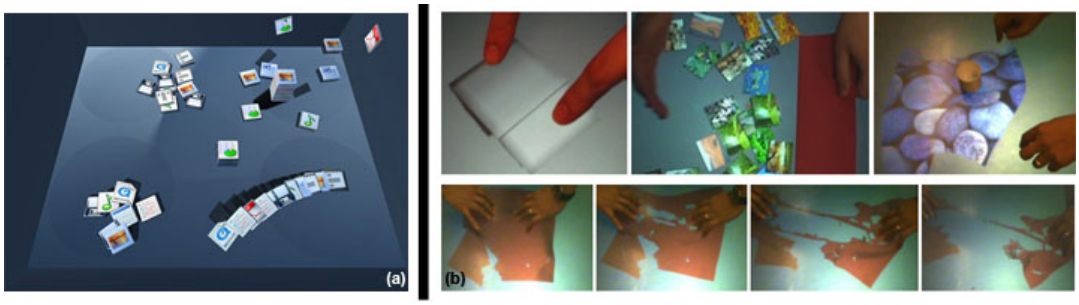


Figure 3: Two examples of physics-based interaction techniques: (a) The Bumptop desktop interface (Agarawala and Balakrishnan, 2006); (b) Physics-based interaction for multitouch surfaces (Wilson et al., 2008)

dragging the files, lifting them and pilling them, as on a physical desktop. The system introduced by Wilson et al. (2008) goes further, by considering contacts, friction and collisions between the fingertips of the user or physical objects he puts on the surface, and virtual objects on a multitouch screen. The user can pin down virtual objects, "push" them to gather them up, and even tear them apart with both hands (see Fig. 3(b)).

In such cases, the user's understanding is instantaneous since his perception of the situation is already highly structured. The functioning of the system and the adapted actions have already been internalized and he has already built adapted psychological artifacts. Nevertheless, the resulting *understandability* will obviously depend on the *quality* of the metaphor —i.e. the similarities between the actual functioning of the instrument and the prior experiences of the user.

Understandability may be favored if the functioning of the instrument is based on activities that the user has perceived. If he has reached secondary intersubjectivity with actors of the corresponding activity, he has also developed adapted psychological artifacts to make sense of it. For example, a non-musician Guitar Hero novice at least has seen guitarists playing their instruments, and implicitly knows the rules of Western tonal music. Therefore, he knows that the device will be operated with simultaneous actions of both hands, which will be structured in time according to the musical structure he knows.

If the user has experienced primary intersubjectivity, he has clues as to what vocabulary of actions is available. Even if this latter kind of experience is less rich, it has the advantage of corresponding to knowledge that is prepared but not yet consolidated. Schemes are less rich, but less constraining regarding assimilation and accommodation. For instance, if an instrument is based on a sign language, a non-expert will know that it will respond to gestures and static hand shapes. Furthermore, the imitation of these gestures has already been

prepared when he has been exposed to such language. But if the vocabulary also includes gestures that do not have any meaning in sign language, the user will not be confused.

Understandability can be evaluated by observing the ability of users to structure their knowledge of the functioning of the instrument, internalize it, and spend more time focusing on goals than on wondering how the instruments works. Corresponding breakdowns occur when the user is unable to understand how to interact with the instrument.

3.6.2 *Operability*

An instrument is *operable* if the user is able to appropriately perform the physical actions that are necessary to use it.

Figure 4: Frédéric Bevilacqua and Julien Bloit from IRCAM using kitchen utensils and a football to play music (Rasamimanana et al., 2011)

Figure 5: The signal processing algorithms designed by Harrison et al. allow to recognize the sound produced when a fingernail is dragged over a surface, as well as the type of surface the user scratches, with a simple piezoelectric microphone (Harrison and Hudson, 2008)

Operability will be maximized for instruments which just require natural gestures, e.g. gaze interaction (Bulling et al., 2012), or gestures that have been automatized as operations, e.g. tangible interaction with everyday objects or the Scratch Input techniques introduced

by Harrison et al. (2008). For example, Rasamimanana et al. (2011) created a set of sensors and software modules to use everyday objects[3] as controllers for music software (see Fig. 4). Scratch Input[4] transforms any surface on which a piezoelectric microphone can be pasted into an interactive surface able to detect when a user drags a fingernail on it (see Fig. 5), as well as several features of the scratching gesture.

In these cases, *operability* is immediate, since the users do not need any practice to perform the gestures. *Operability* might also be favored if the required gestures have already been performed by the user, even if they have not been automatized. If the instrument is not directly *operable* but the associated gestures are feasible, the system must help the user practicing the appropriate gestures. For example, an instrument can require fast gestures, and users may have to practice. Corresponding breakdowns occur when the user is unable to operate the instrument. Furthermore, we propose to consider the comfort of use as a measure of the quality of the *operability* of interaction instruments.

3.6.3 *Learnability*

An instrument is *learnable* if its understanding and operation are guided and if the various levels of expertise are progressively reachable. *Learnability* helps push the limits of the initial *understandability* and *operability* of the instrument. Several visual strategies[5] can help users *understand* which input is taken into account by the system, how to perform it, and what will be the reaction of the system:

- on-line guidance, such as the dynamic guides for mouse strokes provided in the novice modes of Marking Menus (Kurtenbach, 1993) or OctoPocus (Bau and Mackay, 2008), disclosing the complexity of the system in a progressive manner, and guiding both the *understanding* and *operation* of the system by combining feedforward and feedback mechanisms in the context of the application (see Fig. 6(a&b));
- on-line demonstration, such as the animated character designed by Vanacken et al. (2008) who shows how to perform gestures on a multitouch surface (see Fig. 6(c));

3. See a musical performance by IRCAM researchers using the Modular Musical Objects hardware and everyday cookware at www.youtube.com/watch?v=v7_cHlsQaGw

4. See a demonstration of scratch input at www.youtube.com/watch?v=2E8vsQB4pug

5. The example learning strategies presented here are further described in chapter five, in our study on the *learnability* of chording gestures

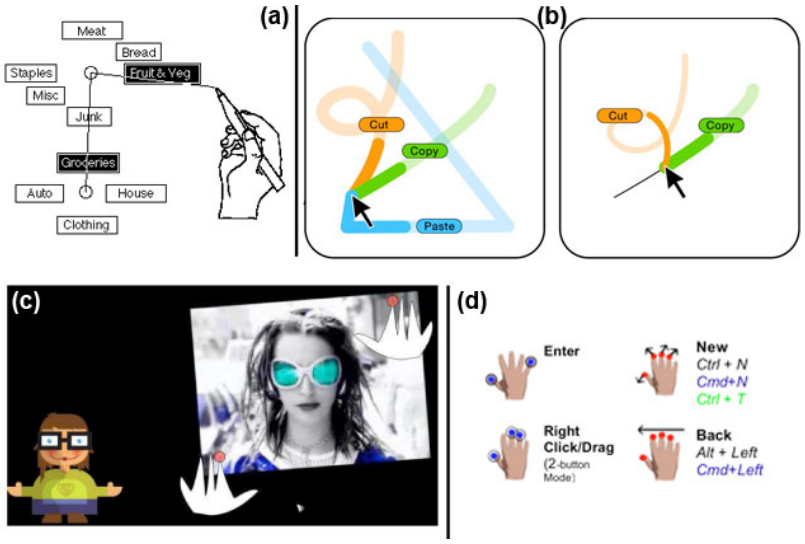


Figure 6: Various *learning* methods for gestural input: the on-line dynamic guides of (a) Marking Menus (Kurtenbach, 1993) and (b) OctoPocus (Bau and Mackay, 2008); (c) the on-line demonstration provided by the animated character designed by Vanacken et al. (2008); (d) the static "cheat sheets" in FingerWorks' (2001) iGesture products manual

– off-line demonstration, such as Apple videos[6] demonstrating their multitouch gestures;

– pictures providing a static summary of the gesture to perform, such as the static "cheat sheets" in FingerWorks' (2001) iGesture products manual (see Fig. 6(d)).

Each of these strategies help users practice the gestures, thus help improving the *operability* of the associated gestural interaction techniques. Nevertheless, only the two first kinds of learning methods — i.e. on-line guidance and on-line demonstration— may dramatically improve the *understanding* of the techniques with little effort from the user. In fact, we have seen in the previous section that complexity can be overcome if it is exposed gradually. Dynamic guides provide progressive guidance when the user performs a gesture, help him understand the differences between the various gestures of the vocabulary, and inform the user of what input has been detected and what is the current state of the system.

Furthermore, in order to ease the transition from novice to expert, syntactic and semantic properties of actions must be preserved and the position of the elements of the interface must be predictable (Scarr et al., 2011), as presented in the previous section. Dynamic guides provided by Marking Menus (Kurtenbach, 1993) and OctoPocus (Bau and Mackay, 2008) minimize the syntactic differences between novice

6. See the video demonstration of Apple multitouch gestures at www.apple.com/osx/what-is/gestures.html

and expert gestures, since the system guides the novice user to progressively perform the gesture, and experts just perform it faster and without guidance. Semantic aspects are also preserved in these cases, since both of these techniques are alternative input methods for existing command vocabularies. In addition, they reduce display demands and emphasize the geometric and spatial properties of the gestures, which might lead to high performance ceiling according to Scarr et al. (2011).

Breakdowns corresponding to *learnability* occur when the user is not guided properly, e.g. when he does not manage to memorize how to interact with the system, when he learns inappropriate or uncomfortable gestures. It can be evaluated by measuring the time spent by users to reach successive significant levels of expertise.

3.6.4 *Expressiveness*

We define four complementary criteria participating in the *expressiveness* of an instrument:

- *semantic width*: the number of actions that are taken into account;
- *semantic variety*: the richness of the differences between these actions;
- *syntactic width*: the number of syntactic rules to combine actions;
- *syntactic variety*: the richness of the differences between these syntactic rules.

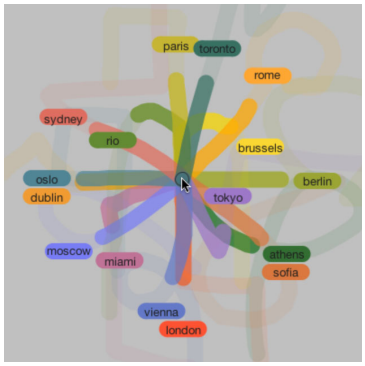


Figure 7: The OctoPocus technique (Bau and Mackay, 2008) is limited in semantic width (here, 16 commands are accessible)

In HCI, the first criteria is often referred to as the *scalability* of an interaction technique. For example, while OctoPocus (Bau and Mackay, 2008) is an efficient dynamic guide for learning mouse strokes, it may be difficult to manage its visual layout with more than sixteen commands, which is defined as a limit in *scalability* (see Fig. 7).

Other projects explored what we term the *semantic variety* of input. For example, Song et al. designed a digital pen augmented with a multitouch sensor: the MTPen (Song et al., 2011). The device avoids

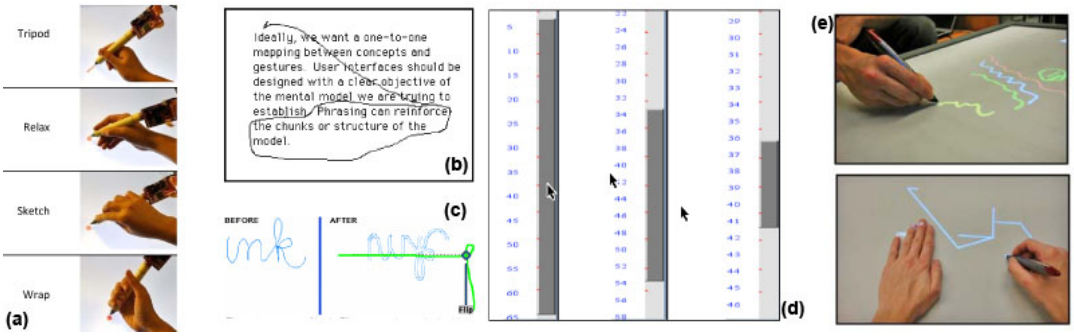


Figure 8: *Expressive* interaction techniques based on the last three criteria: (a) Grips and gestures on a multitouch pen; (b,c,d) phrasing techniques —i.e. temporal composition of actions—; (e) bimanual input —i.e. spatial composition of actions (references are in the text)

auxiliary gestures during interaction and enriches pen interaction by taking into account the number and types of fingers that are grasping the pen, their positions (see Fig. 8(a)), and simple gestures performed on the pen with fingertips, such as double tap or swipe. Therefore, the system takes into account much more features of the gestures than just the pen strokes.

As regards syntax, we distinguish between two dimensions on which actions can be combined: temporal composition and spatial composition. William Buxton introduced *phrasing* as a way to combine sequential actions (Buxton, 1986). He describes *phrasing* as the compound selection of objects, commands and arguments with a single gesture. Fig. 8(b) shows an example provided by the author of a proof-reader's symbol specifying "move", which Buxton describes as the natural way to consider the "move" command: the same line circles the text, specify that it must be moved, and points to the appropriate location. For Buxton, phrasing corresponds to the common way to act and understand things, for instance in language, compound physical movements and music. Therefore, he argues that using phrasing in interactive systems can accelerate the acquisition of expert skills.

Hinckley et al. presented phrasing techniques for multi-stroke pen interaction (Hinckley et al., 2006). In their design, the user presses the button of a digital pen and performs several strokes while keeping the button depressed to select objects, exclude others, select actions and their arguments. When the user releases the button, the phrase ends and the actions are triggered. Fig. 8(c) shows a pen gesture —in green on the figure— combining selection of a word and triggering of a command: selection is done by crossing the word, and the "flip" command is triggered by drawing a pigtail after selecting the word.

Appert et al. (2006) designed the OrthoZoom scroller, with which an orthogonal movement of the mouse while dragging a scrollbar defines the precision —and thus speed— of scrolling (see Fig. 8(d)).

Studies on bimanual input have explored the spatial composition of actions. For example, Brandl et al. designed bimanual pen and direct-touch interaction for multitouch screens (Brandl et al., 2008). With their interaction techniques, putting the hand on the multitouch surface while drawing activates the "straight-line" command (see Fig. 8(e)).

To take a full advantage of this *expressiveness*, the user has to operate the instrument properly and understand its functioning. We can expect a tradeoff between *expressiveness* on the one hand, and *understandability* and *operability* on the other. For example, it will be more difficult to *understand* and *operate* an instrument as the *semantic variety* and *semantic width* increase. The user will have to *understand* the differences between gestures, and learn how to perform them in a very precise manner. Therefore, *expressive* instruments may require efficient *learning* methods that provide precise information about how to interact with the system.

It is important to note that *expressiveness* is not related to the number of actions that are actually used to manipulate objects of interest, which correspond to *effectiveness*. For example, a system can ask a user to learn how to use American Sign Language as an instrument, which is in fact very *expressive*, to trigger a very limited set of commands, which reduces its *effectiveness*.

3.6.5 *Attractiveness*

We define three complementary conditions to the *attractiveness* of an instrument:

- *initial accessibility*: it must provide enough visual and physical affordances to encourage novices engage in first use;
- *initial interest*: users must be aware of the *expressiveness* it will provide them;
- *pleasure of use*: its use must be enjoyable, by avoiding the breakdowns related to the *operability* and *understandability* of the instrument.

Within the opposition between complex and complicated systems pointed out by Norman (Norman, 2010a), complex —i.e. *expressive*— systems can be *attractive* if the power of the instrument is *understandable*.

Understandability	**Result**: Structured knowledge about the actions that are taken into account, and the reaction of the instrument **Requirements**: Accountability (consistency, feedback, eventually abstractions) **Might be favored if the instrument uses**: – Familiar situations; – Situations of secondary intersubjectivity; – Situations of primary intersubjectivity. (depending on the quality of the similarities between the functioning of the instrument and the previous experiences of the user)
Operability	**Result**: Ability to appropriately perform the physical actions that are necessary to use the instrument **Requirements**: Feasible gestures, avoid uncomfortable gestures **Might be favored if the instrument uses**: – Natural gestures; – Gestures which have been automatized; – Gestures which have already been performed.
Learnability	**Result**: Understanding and operation are properly guided **Requirements**: – Features must be progressively presented; – Spatial predictability, low display demands; – Minimize the syntactic differences between novice and expert uses; – Preserve the semantic aspects;
Attractiveness	Initial accessibility, initial interest, pleasure of use
Expressiveness	Semantic width, Semantic variety, Syntactic width, Syntactic variety

Table 1: Summary of the framework for studying the *usability* and *expressiveness* of interaction instruments

Considering these definitions, summarized in table 1, an instrument is accessible to novices if it directly provides them with a minimum level of *understandability* and *operability*. In order to compensate for a lack in one of these two criteria, and to allow novices to become experts, interaction techniques must be *learnable*. An instrument that is fully *understood*, and seamlessly *operated*, is *ready-to-hand*.

Its functioning has been internalized, no breakdowns occur, and the perception and action of the user are structured and instantaneous.

Our proposition to design interaction techniques inspired by expert activities will improve *expressiveness* thanks to the complexity of expert artifacts, *understandability* since knowledge is structured and consistent, and *learnability* if we take advantage of the existing learning methods that have been specifically elaborated. Furthermore, the music-related knowledge and motor skills acquired by non-musicians can make music-based interactive systems both more *accessible* and more *attractive*.

The projects we present in the three following chapters have different positions within that framework. Each of them addresses specific questions and concerns a different field of interaction design.

4

RHYTHMIC INTERACTION: AN EXPRESSIVE AND ACCESSIBLE INPUT METHOD

> *"Musical sensations of a rhythmic nature call for the muscular and nervous response of the whole organism" (Jaques-Dalcroze, Émile and Rothwell, Fred and Cox, Cynthia, 1930, "Eurhythmics, art and education", Barnes)*

As laid out in the previous chapter, rhythm plays an important role in our everyday life. When listening to music, we constantly try to infer the beat, i.e., to perceive regular and salient events, and group events into rhythms (Large, 2001). In fact, we systematically try to perceive rhythm even when none is present (Potter et al., 2009) or when being told to avoid it (Fraisse, 1982). Rhythm is so deeply embedded in our experience of living that we naturally synchronize to periodic events (Leman, 2007), and it can be used to cure some diseases such as stress or sleep disorders (see Sacks, 2008).

While perceiving and reproducing rhythm is recognized as a fundamental human ability by physiologists and neuropsychologists, it is still underused as an interactive dimension in HCI. On the contrary, exploring the spatial dimension of input has been the focus of much HCI research on interaction techniques based on hand postures or gestures, e.g. mouse strokes (Appert and Zhai, 2009), in-air hand postures and gestures (Baudel and Beaudouin-Lafon, 1993), Marking Menus (Kurtenbach and Buxton, 1994). Although some interaction techniques presented in the next section use the temporal dimension of input, more advanced uses of *rhythmic patterns* have received little attention.

We introduce *Rhythmic Interaction* as a complementary way to control interactive systems [1]. It can be used in any event-driven environment for a variety of input modalities: clicking the mouse, hitting keyboard keys or a touch-sensitive surface, moving a motion-sensing

1. This work is the result of a collaborative effort with Guillaume Faure (former Ph.D student at *in|situ|*), his advisor Olivier Chapuis, and my two advisors Stéphane Huot and Michel Beaudouin-Lafon. It was published as a full paper at ACM CHI'12 (Ghomi et al., 2012) and received a Best Paper award (top 1%).

device, etc. However, using such temporal structures to convey information can be particularly useful in situations where the visual channel is overloaded or even not available. Therefore, it has competitive advantages for tactile screens, since it requires less screen space than gestural interaction and no visual attention (Wobbrock, 2009). This chapter presents a first exploration of the design space of Rhythmic Interaction in order to address the following questions:

- *Feasibility*. Even if perceiving and performing rhythm is quite natural, are users able to reproduce, learn and memorize patterns? Can they use them to trigger commands?
- *Interaction design*. The number of possible rhythmic patterns is virtually infinite and they can be presented in several ways. Which patterns make sense for interaction and how to design a vocabulary? What feedback helps executing and learning patterns?
- *Technical issues & Integration*. Like most continuous high-level input methods, e.g. voice, marks, gestures, Rhythmic Interaction relies on a recognizer to segment and interpret user input. How to design effective recognizers that do not require training?

Considering the framework presented in the previous chapter, the *understandability* of Rhythmic Interaction is supported by the psychological artifacts built by non-musicians to deal with rhythm. Their perception of rhythm is highly structured and allow them to make sense of musical syntax. *Operability* is also favored by their deep relationship with rhythm inherited from the internal periodic mechanisms of the human body, and by their ability to reproduce rhythmic patterns by tapping along with music or dancing, acquired from their large exposure to western tonal music. *Learnability* of Rhythmic Interaction will be addressed by comparing the various kinds of feedback that can be provided to users when tapping rhythmic patterns. Advanced learning methods may not be required in this case since Rhythmic Interaction should directly provide satisfying levels of understandability and operability. The few duration values of taps and breaks that we use to build rhythmic patterns will keep Rhythmic Interaction *accessible* to non-musicians, while still leading to a high level of *expressiveness*.

In the following sections, we survey related work and then define a framework for Rhythmic Interaction, narrowing the scope of our study to vocabularies of rhythmic patterns that are relevant in the context of HCI. Then, we evaluate the use of rhythmic sequences of taps performed on a tactile trackpad to trigger commands. We report the results of two experiments that show that (i) rhythmic patterns can be efficiently reproduced by non-musicians and recognized by computer algorithms, and (ii) rhythmic patterns are memorized as

efficiently as traditional keyboard shortcuts when associated to commands. Overall, these results demonstrate the potential of Rhythmic Interaction and open the way to a richer repertoire of interaction techniques. We also describe the recognizers that we created for these two experiments, before drawing some conclusions regarding the design of pattern vocabularies and appropriate feedback.

4.1 USING THE TEMPORAL DIMENSION OF INPUT

The literature in cognitive science has studied the perception, reproduction and use of rhythm from several perspectives: physiology, e.g., perception and action (Glass, 2001; MacDougall and Moore, 2005), knowledge and learning, e.g., language (Petitto et al., 2004), artistic applications, e.g., music (Moelants, 2002), etc. Two major studies on the psychology of rhythm in music are reported by Fraisse (Fraisse, 1982) and by Clarke (Clarke, 1999).

4.1.1 *Existing Interaction Techniques*

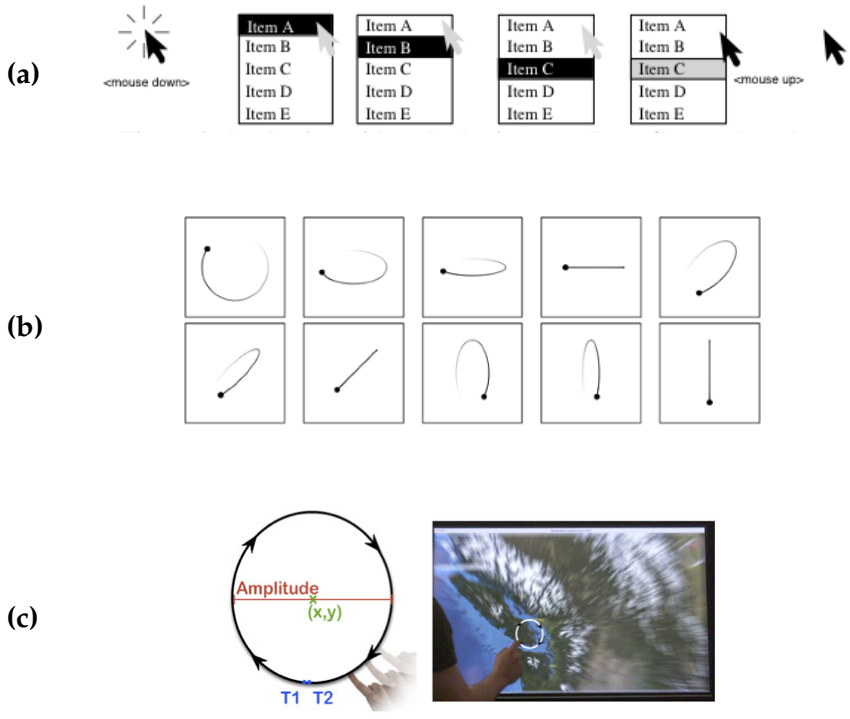


Figure 9: Examples of existing techniques using the temporal dimension of input: Rhythmic Menus (a), Motion Pointing (b), Cyclozoom (c) (references are in the text).

Rhythm is built on the temporal dimension, which is commonly used in interactive software. For example, long clicks are often dis-

tinguished from short clicks to trigger different commands based on temporal criteria. The concept of "dwelling"—freezing the interaction for a short amount of time—is also used to segment gestural interaction (Hinckley et al., 2005) or to explicitly switch mode (Faure et al., 2009). Rhythmic Menus (Maury et al., 1999) successively highlight items at a given rate while the mouse button is pressed. When the user releases the button, the current item is selected (see Fig. 9(a)).

Some techniques are based on the temporal grouping of input events. Double click is the simplest and most common case, but some studies also explored rhythmic motion: Motion Pointing (Fekete et al., 2009) assigns different periodic motions to graphical objects in a scene or items in a pie menu; The user selects the object or menu item of interest by performing the corresponding motion (see Fig. 9(b)). In Cyclostar (Malacria et al., 2010), the user controls continuous parameters, such as the speed of zooming, by performing elliptical oscillatory gestures (see Fig. 9(c)). The rate of the circling motion controls a parameter of the resulting command.

Figure 10: Five-key: A technique using sequences of short and long keypresses to enter any letter with just five keyboard keys (Szentgyorgyi and Lank, 2007)

In the above cases, rhythmic aspects are reduced to periodicity. To the best of our knowledge, only a few techniques involve the reproduction of rhythmic patterns. Five-key (Szentgyorgyi and Lank, 2007) is a text entry system based on rhythmic sequences, where letters can be entered with only five keys (see Fig. 10). However, efficiency and learning were not studied systematically. Crossan et al. use tempo reproduction to select a particular song in a music library by tapping on a mobile device or shaking it citepcrossano6. But relying only on tempo raises some scalability issues that were not assessed. Tapsongs (Wobbrock, 2009) is an alternative to textual passwords where users tap a rhythmic pattern that they have registered with the system for authentication. Finally, Jylhä et al. have recently proposed to use hand clapping to interact with computers (Jylhä et al., 2011). How-

ever, their interaction technique is based on meter, i.e. accentuation of instantaneous periodic events, but does not consider the relative durations of events and interleaving breaks.

4.1.2 *Advantages Of Using Rhythm for Input*

Our approach is somewhat similar to the use of Morse code for encoding characters. However, the design of Morse code was driven by information theoretic issues rather than usability, and while early computers were able to decode human-produced Morse code (Blair, 1959), it has rarely been used in HCI (Chen et al., 2008). Our objective is to propose a comprehensive framework to design rhythmic patterns for interaction, with efficient recognizers that do not need training.

(a)

(b)

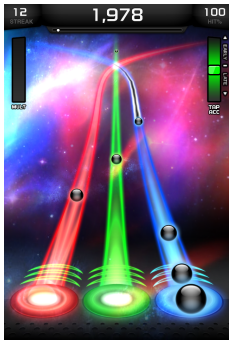


Figure 11: Examples of rhythm-based video games: (a) The note chart and input device of Donkey Konga (Namco Bandai Games), (b) Tap Tap Revenge, a rhythm-based game for mobile devices (Tapulous).

The design of new techniques based on Rhythmic Interaction is not the main focus of our study. However, we have identified a number of potential advantages of using rhythm to interact with computer systems. First, rhythm perception is deeply related to the motor system (Brochard et al., 2008). There is a direct correspondence between

performing a rhythm (action) and listening to a rhythm (potential audio stimulus and feedback). As a consequence, we observe that simple patterns can be reproduced by everyone, as illustrated by the success of popular musical games such as Guitar Hero, Tap Tap or Donkey Konga, where rhythmic structures are recognized and reproduced by non-musicians of all ages. The note charts of Tap Tap and Donkey Konga are similar to the one of Guitar Hero presented in the previous chapter. Donkey Konga uses a special input device with a microphone to sense the sound of a hand clap in addition to sensing when the user hits the device (see Fig. 11(a)). Tap Tap is played by tapping rhythms on the screen of a tactile mobile device (see Fig. 11(b)).

Second, rhythms can be performed in a variety of situations: while performing a rhythm requires as little as a single degree of freedom of one finger, many movements can be performed rhythmically and captured using different sensors, e.g., tapping fingers, tapping feet, or nodding the head.

Gestural interaction typically requires space to perform the gestures, and often interferes with the display space on a small touchscreen. By contrast, Rhythmic Interaction only uses temporal features. Rhythms can be performed on a small area of a tactile device, even in an eye-free context.

Finally, rhythmic structures can be designed in a hierarchical way. By using common prefixes among different patterns, a natural hierarchy emerges that can be internalized by users, facilitating memorization and recall.

4.2 RHYTHMIC PATTERNS FOR INTERACTION

Our definition of a rhythmic pattern comes from music: The elementary structure in music is called a *motif*, which is defined as a "melodic, rhythmic, or harmonic cell" (Manuel, 1960). A *rhythmic motif* represents the temporal structure of a set of notes and consists of the relative durations of notes and silences. Notes and silences can have eight different durations in standard musical pieces, and motifs can contain many notes, leading to a huge number of possible rhythmic motifs.

4.2.1 *Designing Rhythmic Patterns*

Considering the number of commands and actions often used when interacting with computers, such an expressive power is not required. Therefore we propose a restricted definition of *rhythmic pattern* (or simply *pattern*) more adapted to HCI. A rhythmic pattern is

a sequence of taps [2] and breaks whose durations are counted in *beats*. The beat is the basic unit of time and its duration is defined below.

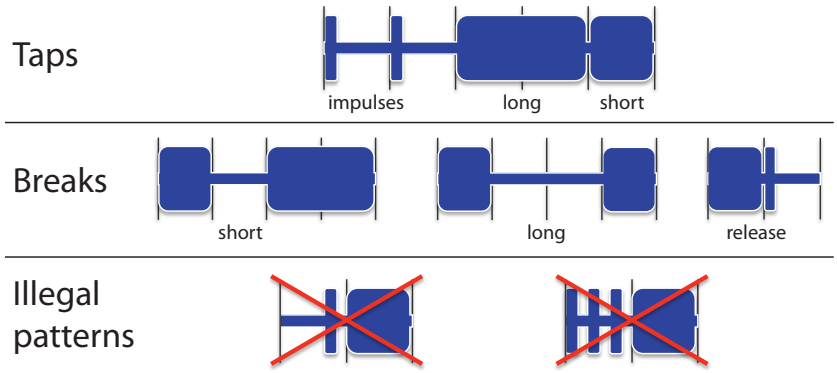


Figure 12: Our rules for defining rhythmic patterns: three types of taps , three types of breaks, and structural rules. Each rectangle represents a tap. The thin gray lines show the beats. The two last patterns are illegal considering our rules, since the first one has an *impulse* which does not start at the beginning of a beat, and the second one has several impulses in one beat.

We define the complete set of possible patterns with the following rules (see Fig. 12):

- Taps can be of three types: *impulse* (a hit on a touch device or a click), *short* tap (one beat) or *long* tap (two beats). A tap starts at the beginning of a beat, and there cannot be more than one tap per beat.
- Breaks can be of two types: *releases* (between two adjacent taps), *short* (one beat) or *long* (two beats). A pattern cannot begin or end with a break, and there cannot be two successive breaks.

This definition of taps and breaks is based on our empirical observation that computer users are familiar with the distinction between instantaneous and long clicks or taps. By adding a third duration and by taking breaks into account, we increase the *expressiveness* of our technique and offer designers more possibilities for selecting a set of patterns among the possible combinations. In comparison to Morse code, we do not need the "intra-character", "inter-character" and "inter-word" breaks that are specific to the coding of language, and we do not allow more than one tap per beat.

The *length* of a pattern is the sum of the durations of its taps and breaks. To simplify reproduction and memorization, we focus on patterns between two and six beats long. The rules above define 5 two-beat patterns, 16 three-beat patterns (Figure 13), 53 four-beat patterns, 171 six-beat patterns and 554 six-beat patterns. By comparison, the

2. The word "tap" reflects our focus on using a tactile device for input in our experiments.

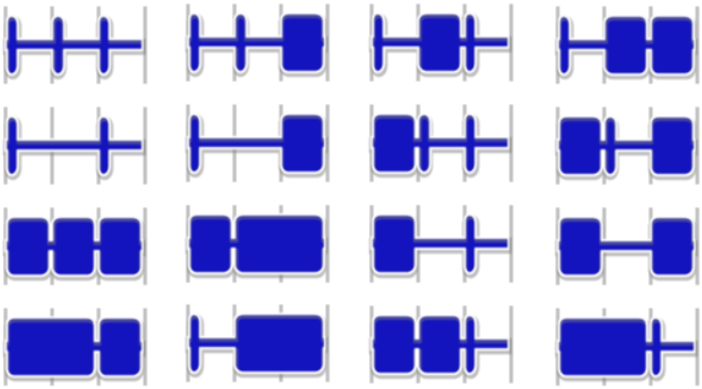


Figure 13: The 16 three-beat patterns defined by our rules.

total number of patterns with n taps is 3^{2n-1}, i.e. 199,290 patterns with up to six taps.

4.2.2 *The Context Of Our Study*

In this entire study, beats occur at the tempo of 120 BPM (2Hz). Thus, the onsets of two consecutive taps are separated by at least 500 ms, i.e. a beat is half a second. This corresponds to a common tempo of human motor actions, e.g. walking (MacDougall and Moore, 2005), and of contemporary music (Moelants, 2002).

As a first step, we only consider rhythmic patterns performed by tapping on a touch-sensitive surface. While keyboards, accelerometers (Lantz and Murray-Smith, 2004; Crossan and Murray-Smith, 2006) or eye blinks (Westeyn and Starner, 2004) can probably be used for Rhythmic Interaction, it is out of the scope of this study. We also do not address the segmentation of patterns from other input. Simple solutions that should be tested include *segmenting in time* by preceding each pattern with a specific short sequence of taps, or *segmenting in space* by performing patterns on a specific location of a device.

A key aspect of this research is to design a recognizer that can reliably identify the patterns produced by users. In a first experiment, we used a *structural* recognizer to assess users' ability to produce patterns accurately. Based on the results, we designed a *pattern classifier* that accounts for user inaccuracies while still discriminating the patterns in the vocabulary. This classifier was used in a second experiment where we assessed users' ability to memorize associations between patterns and commands in an applicative context.

4.3 EXPERIMENT 1: RHYTHMIC PATTERN REPRODUCTION

In order to evaluate the potential of Rhythmic Interaction, we conducted a first experiment where novice users were asked to replicate

patterns presented to them in visual and/or audio form by tapping on a tactile trackpad. Using that device avoids additional mechanicals constraints. The goal was to assess the accuracy of the reproduction and to compare the effects of several feedback mechanisms while performing patterns.

4.3.1 *Recognizer*

The recognizer that we designed for this experiment is based on the above rules for defining the patterns. It first extracts the rhythmic structure as a list of taps and breaks and infers their respective types (impulse, short and long) using autonomous heuristics. The reconstructed pattern is then checked against the vocabulary used for the study.

In order to identify the type of every tap and break in the sequence, the recognition algorithm uses K-means [3] clustering iterated 500 times on duration values. The algorithm builds clusters corresponding to the possible types of taps and breaks: impulse, short and long [4]. A minimum distance of 200 ms between the duration clusters is enforced, corresponding to a maximum tempo for the pattern to be recognized. If two clusters are closer than that distance, they are merged and will be recognized as a single tap type. Thus, if the pattern is performed too fast, taps of different types may be confused by the recognizer. For cluster identification, the reference durations for short and long taps or breaks are set to 500 ms and 1000 ms respectively, and the maximum duration of an impulse or release is set to 180 ms.

After clustering, breaks that correspond to the rest of a beat after an impulse are removed from the reconstructed pattern. The resulting pattern is then looked up in the vocabulary to check if it matches the stimulus provided to the participant.

Note that this recognizer is intentionally very strict, in order to assess the participants' ability to precisely reproduce the patterns. In particular, if the reconstructed pattern is not in the vocabulary, the recognizer will systematically return an error. With minimal knowledge about our definition of rhythmic patterns, the algorithm is able to identify the type of every tap and break in a sequence even in tricky situations, such as when there is just one type of taps. Thanks to clustering, the recognizer adapts to the tempo to a certain extent (the overall tempo can be inferred by comparing the centroids of the clusters.)

3. See the Wikipedia entry on K-means clustering algorithms: http://en.wikipedia.org/wiki/K-means_clustering

4. A break with a duration of an impulse is called a *release*. It occurs when the user lifts his finger from the trackpad between two adjacent taps.

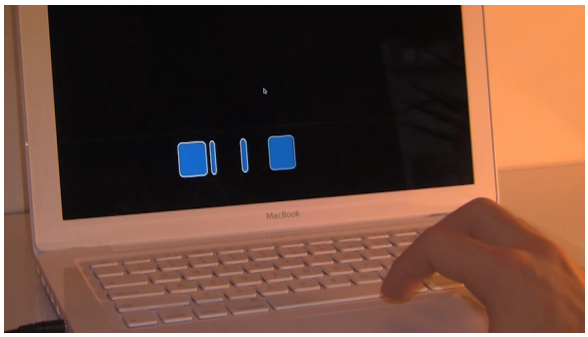
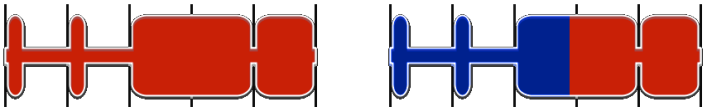


Figure 14: The apparatus and setup of the first experiment.



(a) **(b)**

Figure 15: The stimulus used in the first experiment: A static representation is displayed **(a)** and fills up in synchrony with the audio playback **(b)**.

4.3.2 *Apparatus and Participants*

The experiment was implemented in Java and conducted on a 13" Apple MacBook (Intel processor). Participants tapped the rhythmic patterns on the embedded multitouch trackpad (see Figure 14).

Twelve unpaid volunteers (six female) participated in this experiment, with age ranging from 23 to 53 (mean 29, median 27). Five of them had never practiced music.

4.3.3 *Stimulus*

The pattern to reproduce is presented to the participant with a stimulus combining a static graphical representation of the pattern, visual animation and audio (Figure 15). The visual stimulus is a stationary shape depicting the whole rhythmic pattern, where each rectangle represents a tap (Figure 15a). This shape is then progressively filled (Figure 15b) in synchrony with audio playback. Beats are marked with thin gray lines to visualize the durations of taps.

For audio playback and animation, impulses last 125ms and the tempo is set to 120 BPM or 2Hz (500ms period). This value is a standard in today's popular music music, and is above the "synchronization threshold" measured by Repp (Repp, 2006) for both visual and auditive stimulus, ensuring that participants can perceive and perform it accurately.

The audio stimulus is a 440Hz A, played by the General MIDI Instrument "English Horn" and held at a constant sound level. We

Figure 16: Visual feedback while tapping a pattern.

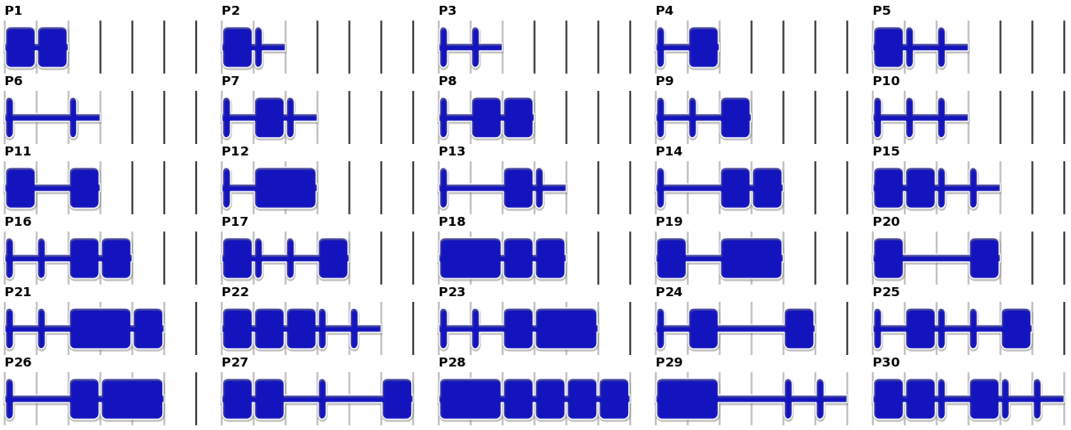


Figure 17: Vocabulary used in the first experiment.

chose this sound as it is soft enough for the subjects to endure during the experiment, but has clear onset and release.

4.3.4 *Feedback*

Participants are presented with four input FEEDBACK conditions while reproducing rhythmic patterns. The Audio feedback plays the same sound as the stimulus as long as the participant is touching the surface of the trackpad. The Visual feedback is based on the graphical representation of the stimulus. The rectangles representing the taps appear dynamically while the subject is tapping on the trackpad (Figure 16). The AudioVisual feedback combines the two previous methods, and there is no feedback at all in the None condition.

The Audio, Visual and AudioVisual feedback methods are expected to help learning, e.g. in novice mode. Conversely, the None condition corresponds to the situation where an expert user is performing patterns in an eyes-free manner without audio feedback.

4.3.5 *Vocabulary*

For this experiment, we selected 30 rhythmic patterns among the 799 patterns with two to six beats generated by the rules described earlier. This vocabulary (Figure 17) contains four two-beats patterns, eight three-beats patterns, eight four-beats patterns, six five-beats patterns and 4 six-beats patterns. We explicitly featured fewer patterns for the extreme situations (two, five and six beats) in order to favor the pattern lengths that maximize the tradeoff between *expressiveness*

and ease of reproduction. Among the patterns with the same duration, we tried to balance the number of taps. For example, for the eight four-beat patterns, two contain two taps, three contain three taps and three contain four taps.

4.3.6 *Task*

A trial consists in reproducing a rhythmic pattern according to the Feedback condition, right after it is presented twice in a row. The participant performs the pattern by tapping on the trackpad with the index finger of her dominant hand. The recognizer then computes the temporal structure of the input and matches it with that of the stimulus. At the end of the trial, the participant is notified about the success or the failure of the match before advancing to the next trial.

4.3.7 *Design and Procedure*

The experiment is a 2×30 within-subject design with factors: (i) Feedback: Audio, Visual, AudioVisual and None; and (ii) Pattern: P1 – P30 (Figure 17).

At the beginning of the session, each Feedback condition is introduced to the participant with a short block of 15 random trials. Then, the participant is asked to perform two warm-up blocks of 15 trials in the AudioVisual feedback condition, which we hypothesize provides the best feedback to become familiar with the task. The three first trials of the first warm-up block are performed by the experimenter to demonstrate the feedback condition to the participant. The second warm-up block is interrupted if the participant reports to be confident enough to start the experiment.

During the main session, measured trials are grouped into blocks according to the Feedback factor. The presentation order for Feedback is counterbalanced across participants with a Latin square. Within each block, the 30 patterns are repeated twice in randomized order. A practice block of 15 randomly selected patterns is performed prior each measured block and participants are allowed to have breaks between and in the middle of each block. Thus, we collected 12 participants $\times$ 4 Feedback $\times$ 30 Pattern $\times$ 2 repetitions $=$ 2880 measured trials. Participants were instructed to be as accurate as possible by paying attention to the discrimination of different types of taps and breaks. Each participant took about one hour to complete the sessions, after which they were asked to rank the feedback methods according to the difficulty of the task on a five-point Likert's scale.

4.3.8 *Quantitative Results*

The overall success rate[5] is 64.3%. This may seem low, but recall that our recognizer is deliberately very strict regarding the temporal structure of patterns, and that it can recognize all 799 patterns with two to six beats, not just the 30 patterns in the study. The precise reproduction of the rhythmic patterns in the study is similar to playing a percussion instrument, a task that musicians can take years to master.

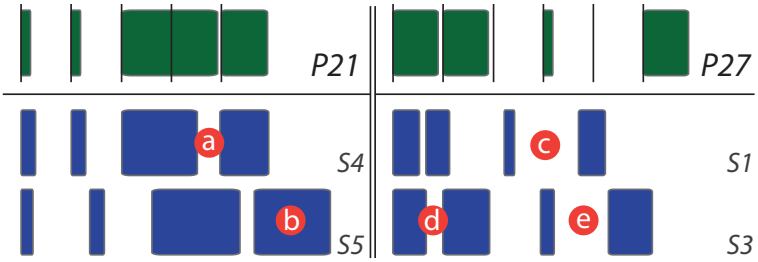


Figure 18: Two patterns (P21 and P27) with reproductions errors by subjects of the first experiment. S4: the last break is too long (a); S5: the last tap is too long (b); S1: the last break is too short (c); S3: the first break is too long (d), the last break is too short (e).

Figure 18 shows typical reproduction errors by study participants, such as release breaks that are too long and recognized as short breaks, or breaks or taps that are too similar to be separated during clustering. Interestingly, errors seem more frequent with breaks than with taps, which is consistent with the finding that users tend to be more precise when performing notes than pauses (Rammsayer and Lima, 1991).

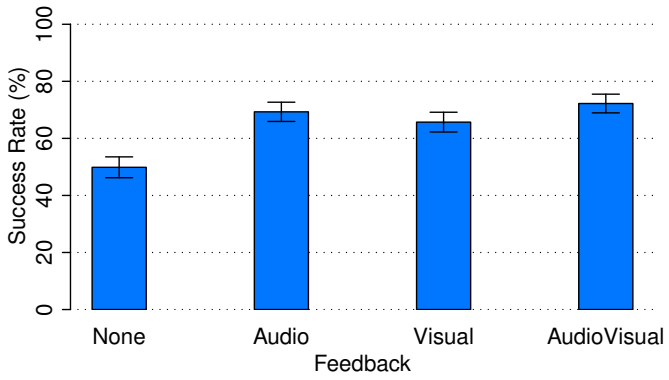


Figure 19: Success rate for each FEEDBACK condition.

A one-way ANOVA for FEEDBACK (with participant as a random variable) reveals a significant effect on success rate ($F_{3,33} = 15.4$, $p < 0.0001$). This effect can be observed in Figure 19[6]. Post-hoc

5. Analysis were performed with SAS JMP Pro: www.jmp.com/software/pro/.
6. In all figures, error bars show the 95% confidence interval.

t-tests with Bonferroni correction show that the None condition is significantly worse than all other feedback conditions. It is not surprising that the absence of feedback while performing the pattern significantly degrades the accuracy of rhythm reproduction.

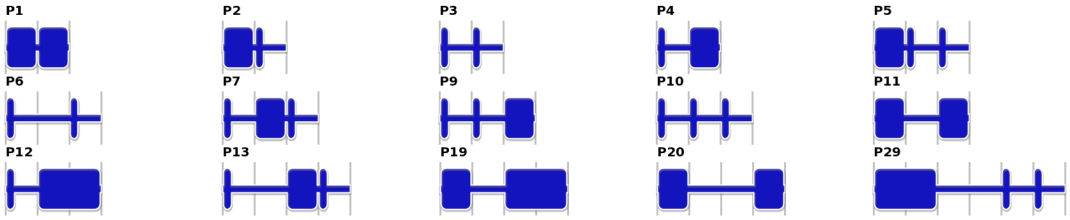


Figure 20: Fifteen patterns having a success rate of at least 70%.

Regarding Pattern, a one-way anova reveals a significant effect on success rate ($F_{29,319} = 25.1$, $p < 0.0001$). We observe a large deviation of the success rate for some patterns: from 16% with P27 to 98% for P10. Fifteen patterns have a success rate of at least 70%: P1–P7, P9–P13, P19, P20, and P29 (see Figure 20). All have at least 3 taps and all but P29 are less than four-beats long. However, some three-tap patterns have a low success rate (below 50%): P14 and P18 (both with 4 beats).

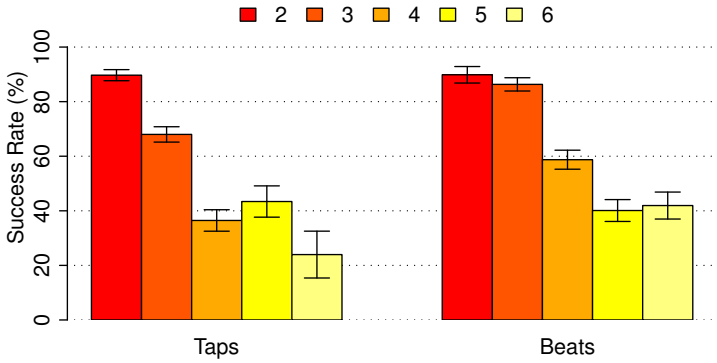


Figure 21: Success rate by number of taps and by length in beats.

We could not identify similarities among the patterns that were difficult to reproduce. However, the number of taps and beats are the most obvious characteristics that can influence the ease of reproduction. In fact, we found a significant effect on success rate for taps ($F_{4,44} = 54.1$, $p < 0.0001$) and beats ($F_{4,44} = 85.5$, $p < 0.0001$), without significant interaction with Feedback. Post-hoc t-tests with Bonferroni correction support this hypothesis since, in most cases, the highest recognition rates were achieved for patterns with a small number of taps or beats (Figure 21).

4.3.9 *Qualitative Results*

Six participants out of 12 preferred the Audio feedback, three the Visual feedback, three the AudioVisual feedback and one no feedback. Moreover, six participants ranked AudioVisual second and 8 ranked the None condition last. Note that many participants pointed out that AudioVisual was confusing, providing too much information. They explained that in most cases, they chose one feedback (visual or auditive) and tried to ignore the other. Half of them preferred the Audio feedback because it was more related to rhythm than graphics. One participant explained that in the None condition, she had to tap a little harder to compensate for the absence of feedback. Since such accentuation and lengthening in duration are often linked in musical performance, this absence of feedback may disturb the precise reproduction of the temporal structure of the pattern, at least in such a situation where the device does not provide tactile or auditory feedback by itself.

We assessed the subjective difficulty of the task with the statement "I found it difficult to reproduce rhythmic patterns". Seven participants disagreed or strongly disagreed, four neither disagreed nor agreed, and only one agreed, but at the same time disagreeing for the None and Visual feedback.

Overall, both quantitative and qualitative results are encouraging and support our hypothesis that rhythmic patterns, as defined by our framework, is a viable input technique for interactive tasks. While quantitative results support the need to provide feedback while performing input, qualitative results provide information about the type of appropriate feedback. Finally, an analysis of recognition errors gives insights on how to create a recognizer that would be more suitable for real applications.

4.4 A PATTERN CLASSIFIER

The goal of the structural recognizer in Experiment 1 was to assess how accurately participants could reproduce a stimulus pattern. This recognizer is deliberately strict, accounting only for variations in the overall tempo of the pattern, and it does not take advantage of the fact that the input patterns are assumed to be part of a given vocabulary. We designed a second recognizer for use in actual applications, that classifies an input pattern against a vocabulary.

In order to recognize a sequence of taps, this *pattern classifier* first counts the number of taps in the sequence and considers the subset of the vocabulary with that number of taps. Then, it calculates a score for each candidate pattern. First, it infers the duration of a beat by considering the duration of the sequence of taps and the number of taps of the candidate. Using this value, it scales the pattern to

match the duration of the input sequence and sums the temporal differences of taps onsets and durations. A duration of a quarter beat is used for impulses and releases between consecutive taps (when lifting the finger from the device). Finally, the score is weighted by the ratio between the inferred beat duration and the 120 BPM reference (500ms).

This classifier is less strict than the structural recognizer because it will always match an input pattern to a pattern in the vocabulary if it is the only one with the same number of taps, unless a threshold is set on the lowest acceptable score. Moreover, normalization makes the recognizer match patterns that are homothetic of each other. This is the reason for weighing the score by the relative beat durations.

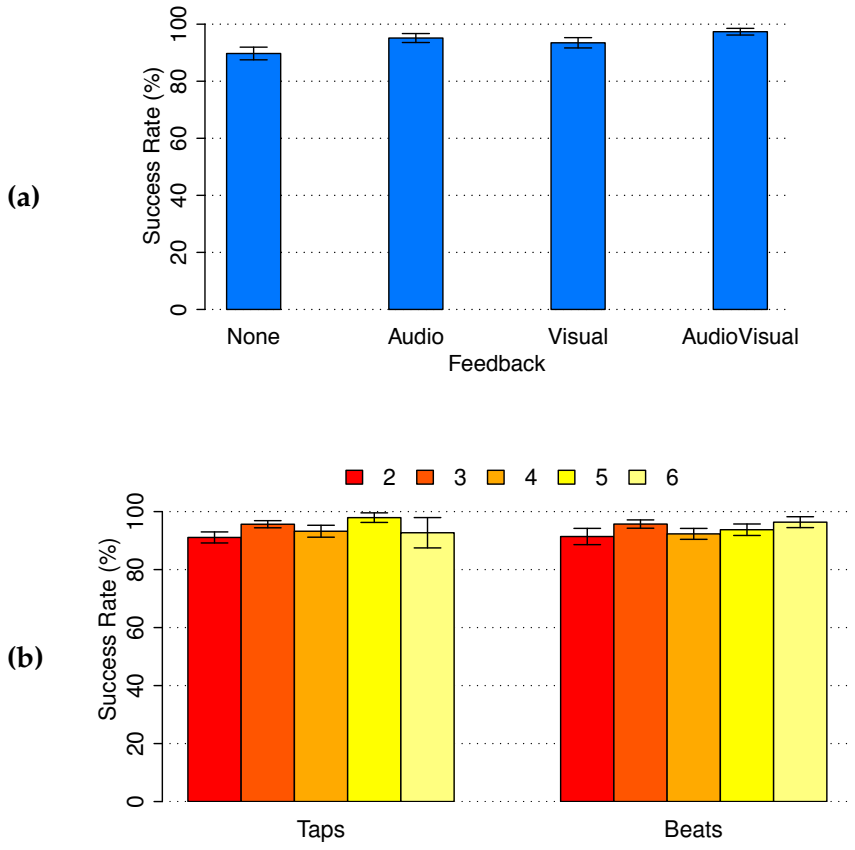


Figure 22: Revised success rate by FEEDBACK (a) and by number of taps and length in beats (b) for the pattern classifier.

We tested this classifier with the data and vocabulary of Experiment 1. The overall success rate rose to 93.9%, more in line with the expectations of an applicative context. As with the previous recognizer, a one-way ANOVA for FEEDBACK reveals a significant effect on success rate ($F_{3,33} = 7.2$, $p = 0.0007$) (Figure 22(a)).

Figure 22(b) shows that unlike the structural recognizer, success rate does not decrease with pattern "complexity": there is no significant effect of the number of taps or the length on success rate[7]. Instead, we observe that success rates are affected by the similarity between patterns: a complex pattern can be recognized quite reliably provided that it is sufficiently different from other patterns with the same number of taps. For example, P30 is the only pattern made of 6 taps in our set, making recognition failure occur only when the subject tapped a wrong number of taps. By contrast, P17 seems to be more "complex" than the "simple" pattern P20 but the former has a 100% success rate and the latter 82%. In fact, the recognizer sometimes confuses P20 with P11. However, a post-hoc t-test with Holm correction reveals no significant difference between patterns for success rates.

In summary, we found that this classifier was well adapted to actual applications. In particular, a designer can create a vocabulary that minimizes the risk of patterns being confused.

4.5 EXPERIMENT 2: RHYTHMIC PATTERNS MEMORIZATION

In order to further validate Rhythmic Interaction, we conducted a second experiment to test whether patterns can be memorized and recalled in order to be used as an alternative to standard techniques for triggering commands. We compared rhythmic patterns with standard hotkeys in a "learn and recall" experiment similar to Appert and Zhai's comparison of gesture shortcuts with hotkeys (Appert and Zhai, 2009), itself inspired by Grossman et al's study of hotkeys (Grossman et al., 2007).

4.5.1 *Variables*

We compare two techniques for triggering commands (TECH factor): `Hotkey` and `Rhythm`. A third condition, `Free`, lets participants choose the technique they prefer.

Each command C_i is a triplet associating an image I_i, used as a stimulus for this command, and two triggering techniques: a rhythmic pattern R_i and a hotkey K_i. The command set (CMD factor) has 14 commands: $C_1, \ldots, C_{14}$. We chose the images symbolizing the commands in a set of common objects and fruits (Figure 23).

For the rhythmic patterns, we selected a representative subset 14 patterns of varying complexity from the vocabulary used in the first experiment, and randomly assigned each pattern to a command. For the hotkeys, we created combinations of a modifier (`Shift` or `Ctrl`)

7. This could be due to the fact that in the vocabulary, there were few patterns with five taps or beats.

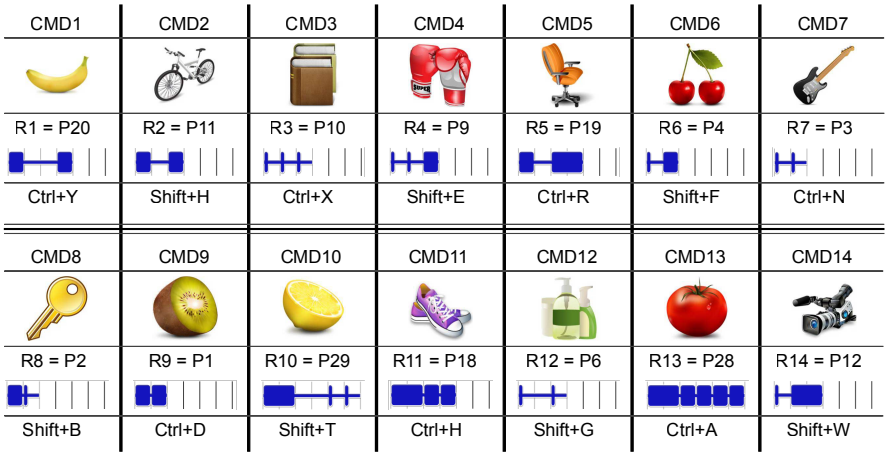


Figure 23: Commands used in Experiment 2. Pxx refers to the patterns of Experiment 1 (see Figure 17).

and a letter. The letters were chosen so that they did not match the first letter of the name of the object representing the command, as in (Appert and Zhai, 2009). The goal is to avoid giving an unfair advantage to hotkeys, since there is no similar mnemonic association between rhythmic patterns and command names. Furthermore, the mapping between commands and hotkeys often varies by application and language. Figure 23 shows the resulting assignment.

4.5.2 *Task*

The primary task of the experiment is to activate a command (C_i), presented by its stimulus image (I_i), with the triggering technique corresponding to the current TECH condition (R_i or K_i). The experiment has two phases: *learning* and *testing*.

During the learning phase, both the image I_i and the corresponding triggering technique (R_i or K_i) are shown to the participant. For rhythmic patterns, the static graphical representation is displayed next to the image (Figure 24a) and the audio stimulus is played twice. Hotkeys are presented with a short animation of the corresponding key-press sequence, also repeated twice, and text (Figure 24b).

In the testing phase, participants are presented with the image I_i only (Figure 24b). According to the current TECH condition, they must perform the corresponding hotkey K_i or rhythmic pattern R_i. If they forgot which trigger to perform, they are strongly encouraged to invoke a help screen by pressing the `Space` key. The task then switches to the learning mode, presenting the shortcut to be performed as described above.

In both phases, the participant must perform the rhythmic pattern or the hotkey. For rhythmic patterns, we use the Audio-only feedback since the first experiment showed that it was effective and par-

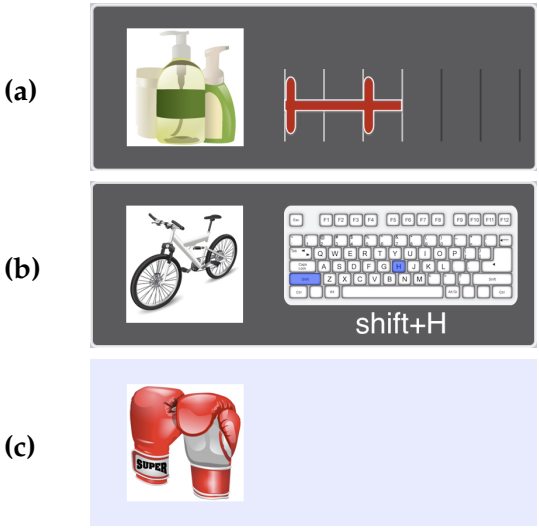


Figure 24: Stimulus in the learning phase for the Rhythm **(a)** and Hotkey **(b)** conditions, and in the testing phase for both conditions **(c)**.

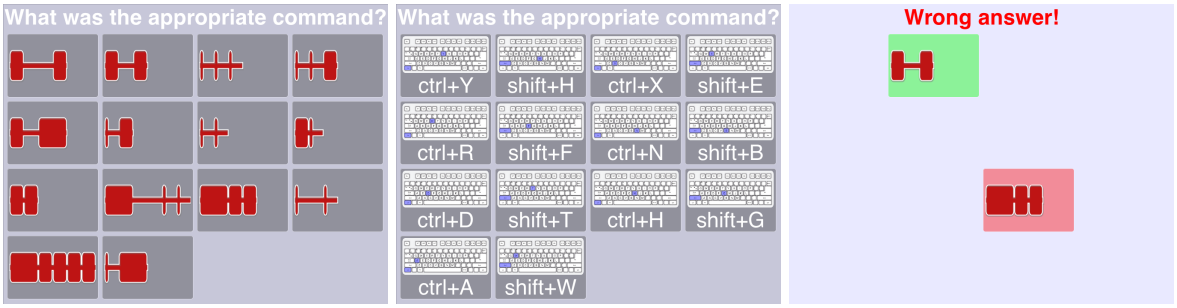


Figure 25: Confirmation in the Rhythm (a) and Hotkey (b) conditions. Feedback for a wrong answer in the Rhythm condition (c)).

ticipants preferred it. Also, this avoids interference with the visual interface. For hotkeys, participants receive the usual kinesthetic feedback while pressing mechanical keys.

After entering each hotkey or pattern, the participants are asked to indicate which trigger they were trying to perform (Figure 25). Then, participants are notified of the correctness of their answer. If the answer is correct, they are given the result of the recognition. If not, the correct trigger is presented before moving on to the next trial. The reason for this procedure is that we are primarily interested in the memorization of the associations, not the participants' ability to perform the triggers. For rhythmic patterns, it also allows us to test the recognition rate of the classifier, by allowing us to distinguish between the system recognizing the wrong pattern and the user performing the wrong pattern.

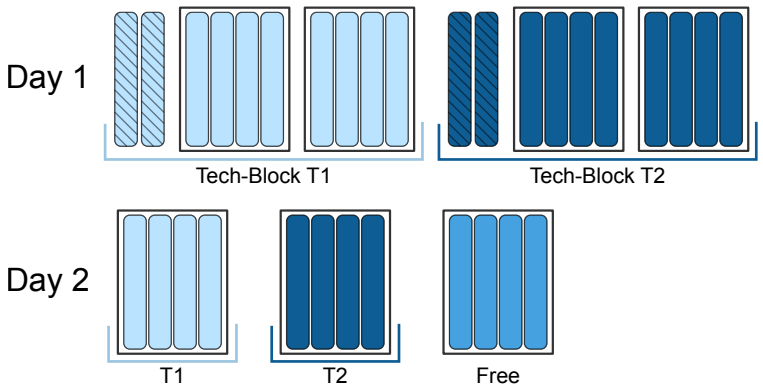


Figure 26: A sample session. Hatched sub-blocks are learning trials, boxed sub-blocks are sub-sessions.

4.5.3 *Apparatus & Participants*

We used the same apparatus as in Experiment 1. We recruited 14 participants (5 female), aged between 22 and 33 (mean 26, median 26). Five of them had participated in Experiment 1.

4.5.4 *Design & Procedure*

The experiment is a within-subject design with technique (Tech) and command (Cmd) as primary factors. The experiment is split into two sessions held on two consecutive days. The first day, all participants are presented with rhythmic patterns in a five minutes practice session based on the first experiment. We use Tech as a blocking factor, counterbalanced across participants. The second day, a Free block is added at the end of the testing phase. In this block, participants can choose to use Rhythm or Hotkey for each trial, but cannot get help.

Each Tech-block is divided into several sub-blocks of 15 trials: (i) two learning sub-blocks with four testing sub-blocks each on the first day; (ii) four testing sub-blocks on the second day. Thus, the testing phase of the experiment is split into SubSessions of 60 trials each: two on the first day to evaluate immediate memorization of triggering commands and one on the second day to test mid-term recall (Figure 26).

In order to simulate a more realistic setup, where some commands are more frequently used than others, we assign an apparition frequency to each of the 14 commands following a Zipf distribution (Grossman et al., 2007; Appert and Zhai, 2009). For the learning phase we use the frequencies (6, 6, 3, 3, 2, 2, 1, 1, 1, 1, 1, 1, 1, 1) and for the testing phase (12, 12, 6, 6, 4, 4, 3, 3, 2, 2, 2, 2, 1, 1). The 14 commands are combined with these frequencies using 7 different permutations, and each frequency assignment is counterbalanced across

participants, resulting in the same number of trials for each command overall. The presentation of the trials is randomized across consecutive pairs of sub-blocks.

The experiment takes about one hour on the first day and 30 minutes on the second day, after which participants are given a questionnaire to collect subjective observations and preferences.

4.5.5 *Quantitative Results*

Our main measures are (i) *recall rate*, the percentage of correct answers in the testing phase without help; and (ii) *help rate*, the percentage of trials where the participants used help in the testing phase. We analyze the results according to TECH and the three sub-sessions of the experiment by considering these measures in the model TECH × SUBSESSION × Rand(PARTICIPANT).

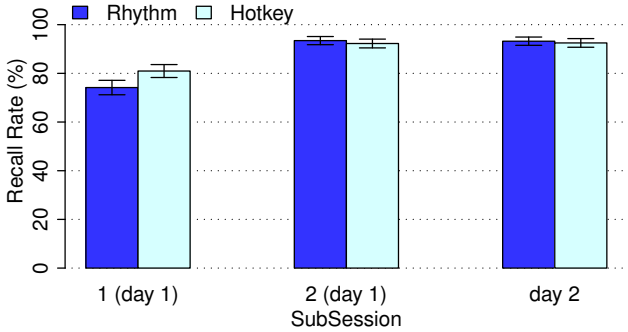


Figure 27: Recall rate for both techniques by sub-session.

We find a significant effect of SUBSESSION on the recall rate ($F_{2,26} = 103$, $p < 0.0001$). A post-hoc t-test with Bonferroni correction shows that the recall rate is significantly lower only between the first sub-session and the two following ones (Figure 27). There is no significant effect of TECH on recall rate ($F_{1,13} = 0.61$, $p = 0.4474$), but the ANOVA reveals a significant interaction effect TECH × SUBSESSION ($F_{2,26} = 5.36$, $p = 0.0113$). Post-hoc t-tests with Bonferroni correction show a significant difference between Rhythm and Hotkey for the first sub-session (74% and 81% respectively). For the remaining sub-sessions, the results are extremely close between the two techniques with a recall rate of about 93% (Figure 27).

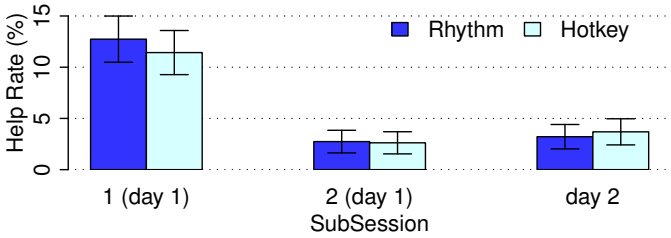


Figure 28: Help usage rate for both techniques by sub-session.

For the use of help, an ANOVA reveals a significant effect of SubSession ($F_{2,26} = 17.3$, $p < 0.0001$), no effect of Tech ($F_{1,13} = 0.04$, $p = 0.8532$), and no Tech × SubSession interaction effect ($F_{2,26} = 0.62$, $p = 0.545$). We find only one significant difference among sub-sessions: help was used more often in the first sub-session than in the two subsequent ones (see Figure 28).

Results for rhythmic patterns and hotkeys are quite similar, suggesting that rhythmic patterns can be memorized as successfully as hotkeys without mnemonics. This is a remarkable result considering how widespread hotkeys are.

Recall rates are consistent across commands. Considering only the Rhythm condition, we build the model Cmd × SubSession × Rand(Participant) for recall rate and help and see a significant effect of Cmd on recall rate ($F_{13,169} = 1.17$, $p = 0.025$). A post-hoc t-test with Holm corrections shows significant differences only between R_3 and R_{13} (recall rate about 97%) and R_{10}, R_{11} and R_{14} (~80%).

To test our classifier, we compare the pattern recognized by the classifier with the answer selected by the participant using the model Tech × SubSession × Rand(Participant). We find a significant effect of Tech ($F_{1,13} = 5.34$, $p = 0.038$), with Rhythm having a significant lower success rate than Hotkey: 85.2% vs. 91.8%. The success rate for Hotkey is surprisingly low, as we expect few if any errors when entering hotkeys. This may be due to participants changing their mind as to which was the right hotkey when they see the answer sheet. For Rhythm, the rate is also lower than expected, but the same phenomenon may have occurred. Indeed, the success rate of Rhythm relatively to Hotkey is 92.8%, close to the rate obtained on the data for Experiment 1 (94%).

4.5.6 *Qualitative Results*

Figure 29 shows the percentage of trials where participants used rhythmic patterns in the Free condition, on the second day of the experiment. Ten participants (out of 14) used rhythmic patterns more often than hotkeys. Seven participants used rhythmic patterns more than 80% of the time, while only one participant used rhythmic patterns less than 20% of the time.

The answers to the questionnaire were generally positive, confirming the previous results. Out of the 14 participants, nine preferred using the rhythmic patterns, three the hotkeys, two had no preference. Those who preferred using rhythmic patterns did so mostly because of the "fun factor" of tapping rhythms, but also because it could be performed "in place" on the trackpad, even for a novice user, without having to visually search the keys on the keyboard. On the other hand, several participants noticed that hotkeys are faster to per-

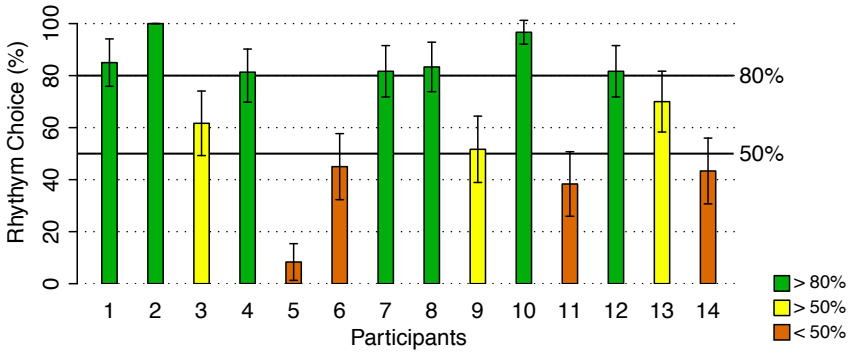


Figure 29: Percentage use of Rhythm by participant (Free condition).

form and preferred to use hotkeys when the corresponding pattern is too long.

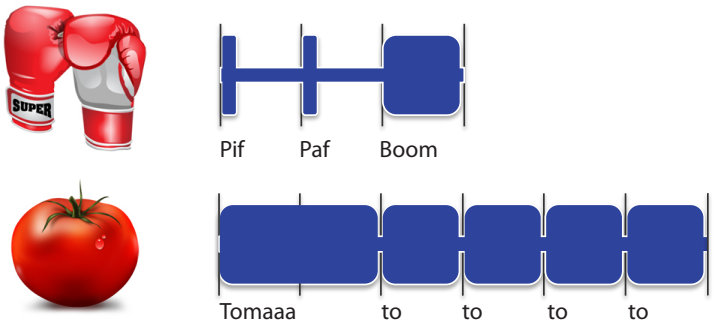


Figure 30: Spontaneous mnemonic strategies reported by participants.

Regarding memorization, some participants reported using mnemonics related to the rhythm itself in order to help memorization (Figure 30). For instance, a subject linked the "boxing gloves" command and the corresponding pattern P9 to a "pif paf boom" onomatopoeia that, for him, echoed the "short short medium" structure of the pattern. Another participant also reported linking the pattern structure with the pronunciation of the object's name, e.g., "toma-to-to-to-to" for command 13 and pattern P28. Subjects also used the graphical representation of patterns to memorize them, which supports our design for this representation. For example, one participant stated that "the rhythmic pattern's visual representation for the cherry looks like a cherry".

These comments suggest that users elaborate efficient strategies for the memorization of rhythmic patterns, based on the rhythm itself or its visualization. Since commands were assigned to rhythmic patterns randomly, we did not expect such associations, but this finding opens the way to studying ways to reinforce these associations. This is commonly done for gestures, e.g., a question mark for help, and hotkeys, e.g. Ctrl-*S* for *S*ave. In particular, various strategies could

be explored to create visual "cheatsheets" for rhythmic patterns or display them next to menu commands, like hotkeys.

In addition, the complexity of performing rhythmic patterns can be turned into an advantage for memorization. Since deeper and greater numbers of levels of encoding and processing help memory (Craik and Lockhart, 1972), combining motor and auditive perception of rhythmic patterns may help users memorize, i.e., encode, their associations with commands.

4.6 APPLICATIONS OF RHYTHMIC INTERACTION

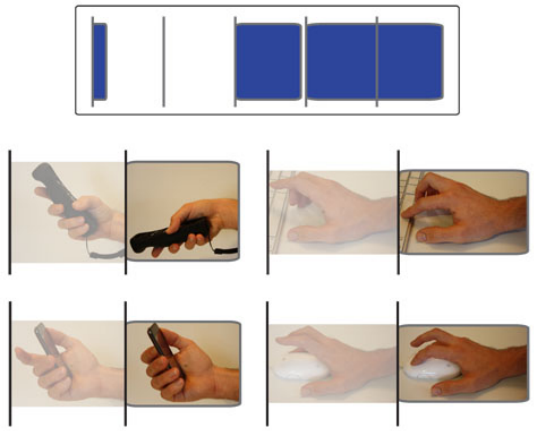


Figure 31: Various devices that can be used for Rhythmic Interaction: rhythmic events can be input by hitting keyboard keys or mouse buttons, tapping or tilting hand-held devices with built-in accelerometers, such as smartphones or the Wii Remote (Nintendo)

Rhythmic patterns are not meant to replace more conventional command input methods. Instead, it is an alternative that may be more adapted to specific situations, such as eye-free operation. It can be used in any event-driven environment for a variety of input modalities (see Fig. 31). It is also a way to enhance existing methods with a richer vocabulary. For example rhythmic patterns could give access to a restricted set of commands such as speed-dialing a phone number, navigating an e-book or switching mode in an application.

In some situations, rhythmic patterns can simplify interaction. For example, bookmarks, menu items or contacts are often organized hierarchically. Rhythmic patterns could match this hierarchy (see Fig. 32). Also, since rhythmic patterns can be performed without visual attention, they can be used with a tactile device in the pocket or while driving, or in the dark, e.g. to shut down an alarm clock, or even with devices that do not have a display.

Rhythmic Interaction also offers novel solutions to well-known problems. Tapping on the back of a hand-held device can be cap-

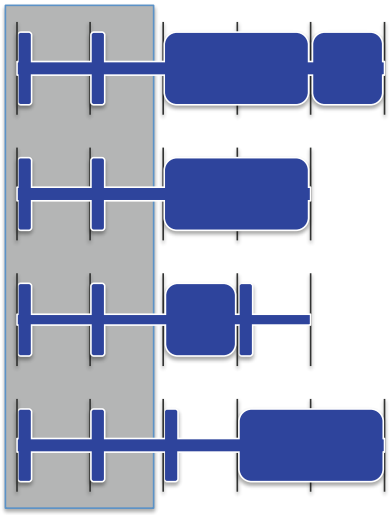


Figure 32: Hierarchical structure of Rhythmic Patterns: These patterns have the same "prefix", here a double tap made of two *impulses*, thus can be matched to command hierarchies such as menus or bookmarks.

tured without extra sensors, thanks to built-in accelerometers or microphones (Robinson et al., 2011). For example, a rhythmic pattern performed while receiving a phone call could add the caller to the contact list, or display extra information such as battery life or signal level. Patterns could also be performed with the non-dominant hand or another part of the body such as the feet (Scott et al., 2010), to switch mode or ignore an incoming call.

4.7 SUMMARY AND PERSPECTIVES

In this chapter, we studied the use of rhythmic patterns in HCI. We explored Rhythmic Interaction as an opportunity to generalize the primitive use of rhythm in existing techniques, e.g., long click and double click, as well as to promote a new input modality. Since Rhythmic Interaction relies on the temporal dimension instead of the spatial dimension used by most input methods, it is well suited when space is limited or when visual attention is not available.

We presented a grammar for creating rhythmic patterns as well as two recognizers that do not require training. A first experiment evaluated the ability of casual users to reproduce rhythmic patterns very precisely with different feedback conditions. We found that some complex patterns can be difficult to reproduce in such a precise way, but that audio and/or visual feedback seems to support *operability* since it improves accuracy. After analyzing recognition errors from the strict recognizer, we designed a chord classifier that reached 94% recognition rate for the 30-pattern vocabulary of Experiment 1. Such

a result validates the *expressiveness* and *operability* of the patterns we designed when used with this pattern classifier.

We ran a second experiment to investigate the memorization of associations between rhythmic patterns and commands, i.e., *rhythmic shortcuts*. The results suggest that rhythmic patterns are recalled as efficiently as traditional hotkeys. The minimum attention given to *learnability* in that study seems sufficient to validate the *understandability* of Rhythmic Interaction since many participants created effective mnemonic strategies to associate rhythms with commands. It means that they have made sense of rhythmic patterns with audio feedback and internalized their use. Such internalization allowed participants to recall 93% of the 14 patterns after the first session of the experiment. The *attractiveness* of Rhythmic Interaction have also been demonstrated, since half of the participants strongly preferred rhythmic patterns over keyboard shortcuts.

This work demonstrates the potential of rhythmic patterns as an input method, and contributes a combination of 14-pattern vocabulary with an audio feedback and an efficient classifier that have proven *usable* by novice users. Beyond triggering commands and switching modes in standard desktop environments, rhythmic patterns could be used in many contexts: eye-free control of a mobile device, such as a cellular phone or a mobile player; remote control of interactive environments such as wall-size displays by tapping on wearable sensors without the need for visual attention; selection of an object on a tabletop when it is not easily reachable, etc.

Future work on Rhythmic Interaction can address issues such as the segmentation of patterns, the scalability of the vocabularies and the speed of execution. Another area for future work is the use of multiple fingers or both hands to tap patterns and to combine Rhythmic Interaction with other interaction techniques. More complex actions than tapping should also be explored to enter rhythmic structures, such as performing sequences of gestures or keyboard taps, as well as the use of the temporal dimension to convey additional information. Furthermore, rhythmic *output*, such as vibration patterns on mobile devices, seems worth studying since perception and performance of rhythmic patterns are tightly linked. Existing studies show that rhythmic patterns can be efficiently perceived via purely tactile stimulation (Brochard et al., 2008).

5

ARPEGE: MAXIMIZING EXPRESSIVENESS AND IMPROVING LEARNABILITY OF CHORDING GESTURES

"If it has more than three chords, it's jazz." (Lou Reed)

Mastering piano chords requires expertise acquired over time through extensive practice. Nevertheless, a music novice can easily learn a restricted set of chords to play along with a song (Lahav et al., 2005). Similarly, non-musicians have also internalized and automatized some multi-finger gestures such as keyboard shortcuts or gestures on multitouch trackpads. These experiences should support the *operability* of chord-based interaction, and the methods used for teaching piano can give us insights into new ways to overcome the complexity of multi-finger gestures.

These teaching methods include learning chord vocabularies, studying differences and conflicts among chords, and practicing with appropriate hand posture. But even if non-musicians have an intersubjective knowledge of musical chords acquired from listening to music and observing instrument playing, they are not used to learn *expressive* vocabularies of chording gestures. Therefore, the *understandability* of chord-based interaction may be limited and must be compensated by a high level of *learnability*.

Today, *multitouch* devices have become widespread and multi-finger interaction is a new rich input method that is becoming a fundamental part of the user experience. New operating systems feature small vocabularies of multi-finger gestures, and ad-hoc gesture sets have been presented in the literature. However, the design of *expressive* vocabularies of chording gestures has received little attention. Furthermore, few studies have taken into account the *comfort* of use of chording gestures, and no learning method has been particularly designed for *expressive* vocabularies of chording gestures.

In this chapter, we study *multi-finger chording gestures* on multitouch screens, where several fingertips are laid on the surface at the same time and then lifted simultaneously, with no additional movement performed.

When performing gestures to interact with the computer, users not only have to learn how to perform the gestures properly —i.e., what the system can recognize—, but also the possible commands —i.e., the available vocabulary. *Dynamic guides*, such as the novice mode of *Marking Menus* (Kurtenbach, 1993) or *Octopocus* (Bau and Mackay, 2008), presented in the next section, have proven efficient to address these problems while minimizing visual complexity by guiding the user through gesture performance via progressive *feedforward* and *feedback*. This strategy seems particularly adapted to multitouch screens where input and output are co-located. However, it has not yet been applied to large vocabularies of chording gestures.

In order to improve the *learnability* of chord-based interaction in this way, we introduce *Arpège*, a contextual dynamic guide inspired by piano teaching. The goal of *Arpège* is to simplify learning chords by breaking down their complexity: fingers are laid down one after the other, while *Arpège* provides feedback, feedforward, and guides eventual position corrections. The goal of *Arpège* is not only to teach chords, but also to allow novice users to explore the vocabulary, making chording gestures more *accessible*. Finally, it should improve the transition from novice to expert, since novices and experts will perform the same gestures at different paces, and *Arpège* takes advantage of partial memorization.

In the following sections [1], we survey related work and then propose simple guidelines based on studies of the motor abilities and biomechanical constraints of the human hand for creating *expressive* chord vocabularies that maximize *operability*. We assess our guidelines in a first experiment by evaluating perceived comfort and *understandability* of a representative chord set. Chords with only fingers in their "relaxed" positions are considered easier than others, and chords involving more fingers are more difficult.

Next, we present our chord recognizer that does not require training from the user, and describe *Arpège*. In a second experiment, we compare *Arpège* with a *cheat sheet* — the common static chord learning method — when learning chords as command triggers. The memorization rates with the two techniques are not significantly different, but *Arpège* exhibits some advantages regarding *learnability*: it is adapted to users' spontaneous strategies to *understand* chords, and helps users build mnemonics. Finally, we draw some conclusions on the difficulty of performing and learning complex chords.

1. This work was initiated with Olivier Bau (former Ph.D student at *in|situ|*) and his advisor Wendy E. Mackay. The contributions presented in this chapter are the result of a collaborative effort with my two advisors Stéphane Huot and Michel Beaudouin-Lafon. An early version has been presented at a workshop on tactile interaction (FITG 2010 (Ghomi et al., 2010)) and a full paper is currently in preparation for submission.

5.1 USING AND LEARNING CHORDING GESTURES

5.1.1 *Multi-Finger Chords*

Figure 33: Engelbart demonstrating the original "keyset" in 2010 (Photography by Evan Schaffer)

As early as the 1960's, Engelbart et al. presented a five-finger chording keyboard for expert computer users (Engelbart, 1962). Their *keyset* is used with the non-dominant hand (see Fig. 33), while the dominant hand acts on the mouse or the keyboard. While such chord-based interaction techniques have existed for decades, gestural interaction techniques for multitouch surfaces mostly combine placement and movement of fingertips and other parts of the hand in contact with the surface. For example, Freeman et al. define a taxonomy of multi-finger and whole hand static and dynamic gestures (Freeman et al., 2009), reused by Bragdon et al.(2010).

Chording gestures, however, have shown to be *efficient* when used for direct command triggering or in multi-finger menu techniques, as presented in the next section. In addition, chords have the advantage not to interfere with dragging on a multitouch surface, contrary to other multi-finger gestures, as pointed out by Bailly et al. (2008).

5.1.2 *Designing Chord Vocabularies*

Several researchers have addressed the design of multi-finger gesture by asking casual users to create gestures by themselves, in order to create *accessible* vocabularies. Morris et al. (2010) reused Wobbrock et al.'s methodology (2009), who propose to design gestures "by first portraying the effect of a gesture and then asking users to perform its cause". The resulting gestures are easy to perform without training, and may maximize both *understandability* and *operability*. However, such approaches offer a limited range of *expressiveness* by favoring an immediate use with no possible transition from novice to expert.

Furthermore, the resulting gestures do not take full advantage of the physical capabilities of the hand, and might be influenced by the common multi-finger gestures shown in advertisements for multitouch devices, which have not been validated as *expressive*, *effective* or *comfortable*.

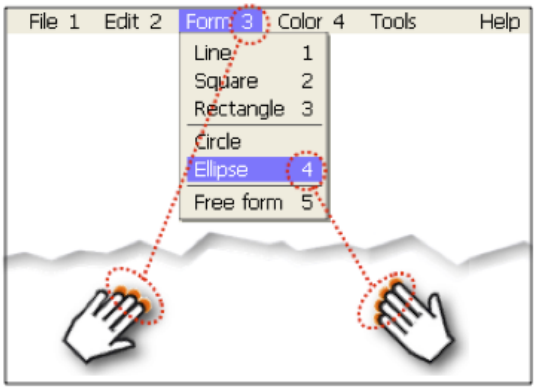


Figure 34: The FingerCount chording technique (Bailly et al., 2010): the user navigates in two-level hierarchical menus by touching the surface with fingertips of both hands. The number of touching fingers of the left hand selects the first level menu entry. The second level menu entry is then selected with fingers of the right hand.

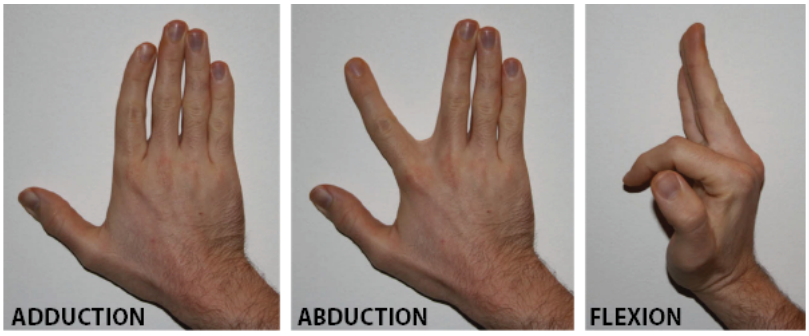


Figure 35: Performing the three common finger movements with the index finger: *adduction*, *abduction* and *flexion*

In fact, chording interaction techniques for multitouch surfaces rarely consider the relative positions of fingers. For example, the FingerCount technique (Bailly et al., 2010) allows users to navigate two-level hierarchical menus by touching the surface with a certain number of fingertips of both hands (see Fig. 34). Therefore, the hand can be kept in a "relaxed" posture, since only the number of fingertips touching the surface is taken into account. Although these approaches can minimize muscle tension in the hand, we argue that chord-based interaction can take advantage of the three common finger movements —*flexion*, *adduction* and *abduction* (see Fig. 35)— while remaining *operable*.

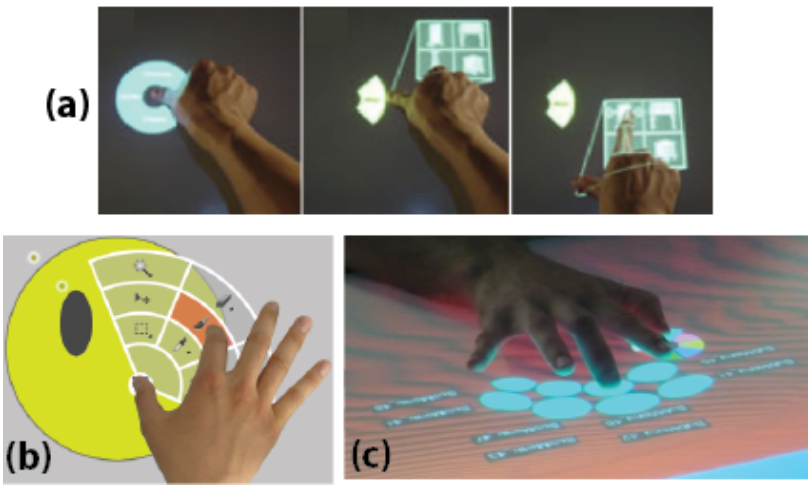


Figure 36: Multi-finger interaction techniques using finger movements: (a) the Furniture Palette (Wu and Balakrishnan, 2003); (b) the Multi-Finger Pie Menu (Banovic et al., 2011); (c) the MultiTouch Menu (Bailly et al., 2008)

Examples of multi-finger interaction techniques using these movements are presented in figure 36. With the Furniture Palette (Wu and Balakrishnan, 2003), the user "double taps" the surface with his thumb and slides it on one of the four items of a Marking Menu (Kurtenbach, 1993). This action invokes a squared toolglass (Bier et al., 1993) —i.e. a semi-transparent contextual menu— which is "attached" to the thumb (see Fig. 36(a)). A second finger — the index finger in the figure— is then used to operate tool selection in the toolglass.

Similarly, the Multi-Finger Pie Menu (Banovic et al., 2011) is a two-finger menu where the second finger can take nine different positions relatively to the position of the first finger in order to select an item (see Fig. 36(b)).

The MultiTouch Menu (Bailly et al., 2008) takes into account eight positions for the thumb, two positions for the middle and ring fingers, and up to four positions for the index and little fingers (see Fig. 36(c)). The position of the thumb selects the first level of the menu, then any other finger selects the second level. Then, a new circular menu appears where the second-level item has been selected. An additional finger movement is performed to operate item selection in this third leve of the hierarchical menu. In this study, the authors address the question of comfort of use by taking the mechanics of the human hand into account.

But in the end, the MultiTouch Menu requires the user to lay the palm of the hand on the surface, which constrains finger movements. Furthermore, although these techniques are *expressive*, by considering several positions for each finger and by allowing users to navigate in

hierarchies of commands, all of them are based on menus and require the user to perform several finger movements, unlike chording gestures.

5.1.3 *Learning Chords*

Even if experts can efficiently perform chording gestures and we can design *accessible* chord vocabularies, we still face the problem of how to help users move from novice to expert performance, i.e., *learnability*. As Rekimoto et al. point out in their article on PreSense (Rekimoto et al., 2003), users face a steep learning curve and need effective guidance to learn large vocabularies of finger combinations. In this section, we further discuss some learning methods already presented in section 3.6.4.

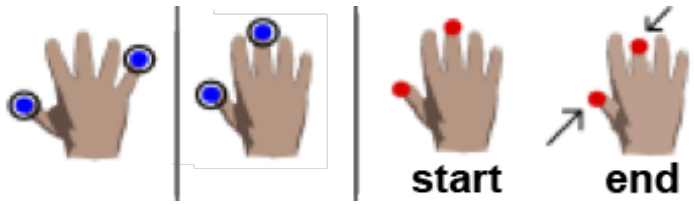


Figure 37: FingerWorks' "cheat sheets" (2001): pictures provide a visual summary of the actions to perform for both static —in blue— and dynamic —in red— gestures.

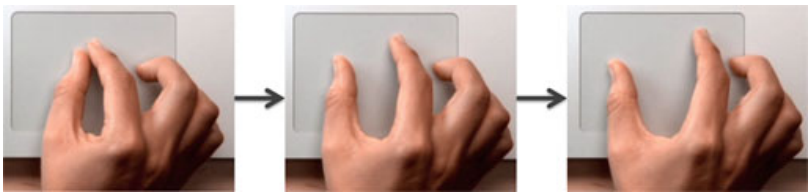


Figure 38: Apple's videos for learning multi-finger gestures.

Some precursor commercial multitouch systems offered only *offline*[2] learning techniques made of pictures summarizing the gestures to perform, such as the "cheat sheets" accompanying FingerWorks' (2001) products (see Fig. 37). The main drawback of these approaches is that the user has to switch context from the application to the *offline* help system. Furthermore, while "cheat sheets" can provide a visual summary of the entire chord vocabulary, pictures do not provide any information about *how* to perform the gesture, which may make complex chords difficult to *understand* and reproduce.

Conversely, Apple's multi-finger gestures are taught in a dynamic way: movie clips demonstrating every gesture are available

2. What we refer to as *offline* learning techniques are methods displayed in a different context than the one the user is working in. It requires the user to switch to another application, watch a video or read instructions on paper.

in the device configuration tool (see Fig. 38). One step further, TouchGhost (Vanacken et al., 2008) shows a two-handed character demonstrating a gesture while the system simulates its result directly in the context of the application (see Fig. 6 in section 3.6.3). These last two systems provide dynamic illustrations of what to do but do not guide the user when performing gestures. They require the user to watch the whole animation, to interpret it and to reproduce the gestures from what he understood.

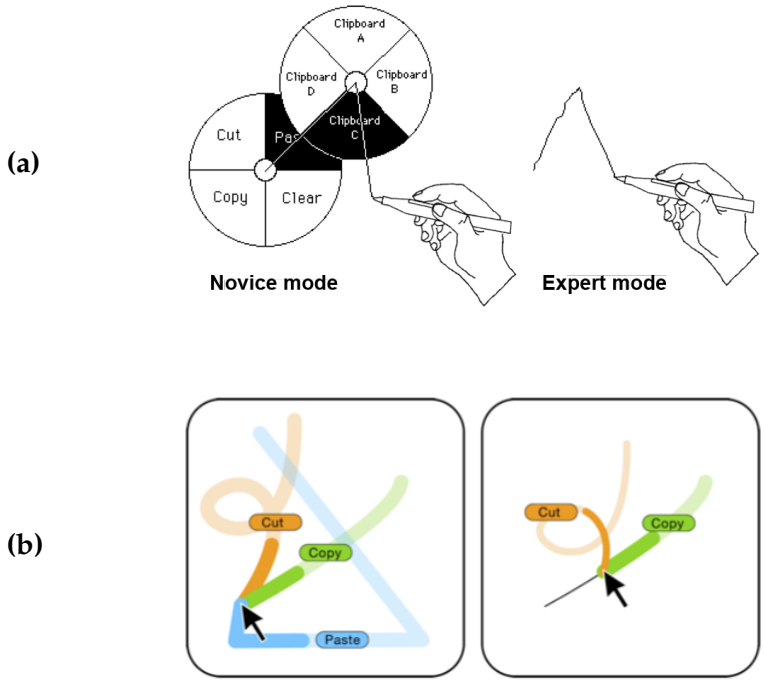


Figure 39: Dynamic guides for learning pen and mouse strokes: (a) the novice and expert modes of the Marking Menus (Kurtenbach, 1993); (b) the novice mode of OctoPocus (Bau and Mackay, 2008)

On the other hand, "dynamic guides" are *online* gesture-learning systems that maximize *learnability* by providing continuous guidance in the context of use. They allow users to get rid of deliberate practice in offline applications and to learn and memorize gestures implicitely while following the system. They provide a combination of: (i) guidance via progressive *feedforward*, to help users perform gestures properly and explore the vocabulary of commands; and (ii) *feedback* to let them know whether their gestures are properly recognized or not.

For instance, Marking Menus (Kurtenbach, 1993) are hierarchical menus in which the user navigates by performing a sequence of pen or mouse strokes. Its novice mode helps the user explore the menu and learn the appropriate sequences of strokes, by progressively exposing the levels of hierarchy (see Fig. 39(a)). When the user knows

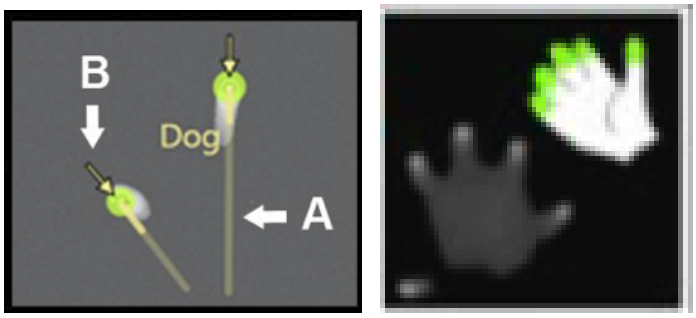


Figure 40: ShadowGuides (Freeman et al., 2009): a deported dynamic guide for learning multi-finger gestures, which shows the start position and direction of movement for dynamic gestures (on the left), but is similar to "cheat sheets" for chording gestures (on the right, the shadow of the hands shows five points, and fingertips are highlighted on the drawing)

which sequence of strokes he has to perform to reach the desired item, he no longer needs guidance, and can perform it directly.

Similarly, OctoPocus (Bau and Mackay, 2008) is a dynamic guide for helping users learn vocabularies of mouse gestures associated to commands. When the user invokes OctoPocus, the system displays the beginnings of all the gestures. Then, while the user follows the label of the command he wants to trigger, the labels of the other commands progressively disapear, indicating that they are no longer reachable (see Fig. 39(b)). The system follows what the user is doing, and can adapt, to a certain extent, its guidance to his particular way of performing the gesture.

The main benefit of dynamic guides is that they improve *understandability* and permit a smooth transition from novice to expert (Bailly et al., 2008; Banovic et al., 2011; Kurtenbach, 1993; Lepinski et al., 2010), since the gestures that novices perform while following the guide are identical to those performed by experts. Experts only perform the gestures faster and without guidance.

Only a few dynamic guides have been designed for multi-finger gestures. ShadowGuides (Freeman et al., 2009) provide efficient guidance to finger and hand movements by showing "the user shadow expected by the system", as well as the direction of the movement and the eventual evolution of the contact shape [3]. However, for chords, it is restricted to a deported "registration pose guide", similar to *cheat sheets* (see Fig. 40). These thumbnails show which fingers are involved in the gesture but still require the user to interpret *how* to perform it.

Gesture Play (Bragdon et al., 2010) is a dynamic guide for multi-finger gestures based on "fun" and physical metaphors including spring widgets, wheel widgets and button widgets to motivate the

3. See a video of ShadowGuides at http://www.youtube.com/watch?v=0faNH05q38s

Figure 41: Gesture Play (Bragdon et al., 2010): a dynamic guide for multi-finger gestures based on widgets

users to learn and rehearse the gestures. For chording gestures, the system provide a step by step "button widget affordance" (see Fig. 41).

However, these two techniques only consider one possible position for each finger, and do not allow the user to explore the whole vocabulary, unlike OctoPocus (Bau and Mackay, 2008). With our rules for chord design and *Arpège*, we respectively address these two problems of *expressiveness* and *learnability*.

Our first objective is then to define guidelines for the design of larger chord vocabularies that are not restricted to menu selection, unlike the techniques presented in the previous section. These guidelines must respect biomechanical constraints of the hand in order to maximize *operability*. We also address the use of chord vocabularies in practical applications (e.g., to invoke commands) with the design and evaluation of the *Arpège* technique, which aims to improve *learnability* and *accessibility*.

5.2 DESIGNING CHORDING GESTURES

We define a *multi-finger chording gesture* as the action of placing two or more fingertips on a touch surface and lifting them simultaneously. We consider not only which fingers are involved in the chord, but also their relative positions that can involve flexion, adduction and abduction (see Fig. 35). In this section, we point out that fingers have different motor capabilities and are not equally able to move independently of each other: some fingers are weaker, such as the little finger, others are stronger, such as the index finger; some can move more freely, such as the thumb, others have their movements constrained by the neighboring fingers, such as the ring finger.

We present two guidelines for designing *expressive* vocabularies of chording gestures, which take into account the mechanics of the hand: the first one avoids some uncomfortable finger combinations, the second one defines additional finger positions to extend the design space

of chord vocabularies. By maximizing comfort of use, we aim to make chord-based interaction easily *operable* and compensate for the preference novices can have for user-defined gestures, as reported by Morris et al. (2010) and Wobbrock et al. (2009).

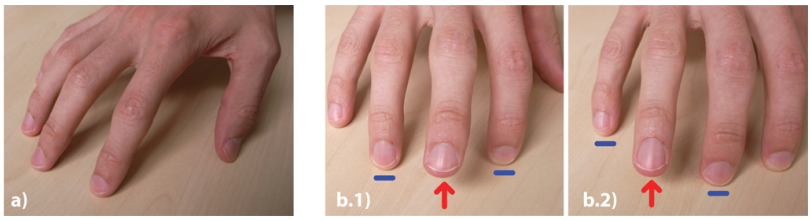


Figure 42: a) Relaxed hand posture on a flat surface. b) Lifting the middle or ring fingers is uncomfortable.

Using Baudel et al.'s terminology (Baudel and Beaudouin-Lafon, 1993), we first consider the "*relaxed*" position of the hand on a multitouch surface. Similar to the reference position for playing piano, the palm is parallel to the surface, all fingertips touching it, and all the fingers being slightly curved (see Fig. 42(a)). Fingers are neither too close to each other, nor too much spread apart, which minimizes tension in the hand. By lifting one or more fingers from this relaxed position, but excluding configurations with only one finger touching the surface, we create $2^5 - 1 - 5 = 26$ possible finger combinations.

5.2.1 *Mechanical Constraints For Finger Combinations*

In order to identify *comfortable* finger combinations among these, we must take into account that fingers cannot all flex or extend independently of each other. Lee et al. (1995) measured the interdependence of fingers during flexion and extension, and specified the angular constraints of the joints between the fingers and the palm. Their study can be used to quantify how easy or hard it is to lift each finger independently. The authors report that the middle and ring fingers have the strongest constraints during flexion. As a consequence, keeping any of these fingers lifted without also lifting its neighboring fingers requires an important muscular effort (see Fig. 42(b)), leading to our first guideline:

Avoid chording gestures where either the middle or ring fingers are lifted while the neighboring fingers are touching the surface.

Following this guideline, eight of the 26 configurations may be uncomfortable. But it leaves us with 18 a priori comfortable chords based on the relaxed position (see Fig. 43).

Figure 43: The 26 possible finger combinations or "relaxed" chords. Eight might be uncomfortable according to our first guideline.

5.2.2 *Additional Finger Positions*

Thanks to the degrees of freedom of the hand, fingers that are involved in a chord can also be moved to various positions: they can be flexed or spread apart on the surface (*abduction* when spreading the fingers apart, away from the centerline of the hand; *adduction* when closing the fingers together, pulling them toward the centerline of the hand). But natural constraints of the hand restrict these movements.

First, lateral movements of the middle and ring fingers are limited by the two neighboring fingers. Second, Lang et al.'s studies on mechanical coupling of fingers (Lang and Schieber, 2004) show that fingers can be ordered by their level of independence, from best to worst: thumb, index finger, little finger, middle, and ring. These limitations due to linkages between tendons have also been reported in studies in the field of music, especially concerning the ring finger (Watson, 2006).

We use these observations to extend the design space of chording gestures by defining additional positions for the most independent fingers, i.e., the thumb, index finger and little finger. Therefore, our second guideline is:

Create additional chords by flexing fingers or moving the most independent fingers (thumb, index finger and little finger) sideways.

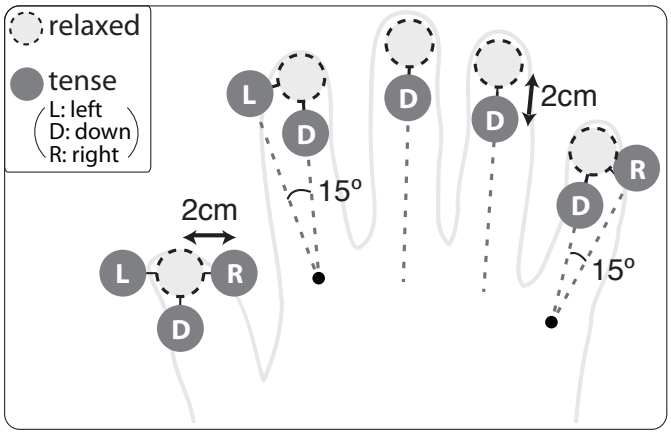


Figure 44: Possible finger positions taking advantage of the degrees of freedom of the hand. Light grey dots indicate "relaxed" positions, dark grey dots are the additional "tense" positions.

Figure 44 shows the additional positions we define for each finger according to this guideline. For the thumb, which is the most independent finger, we define three additional positions: down, left and right, each position being 2 cm from the relaxed position. All other fingers can be flexed to reach a position which is 2 cm below the relaxed position on the axis of the finger. This distance corresponds to the standard size of keys on computer keyboards according to studies in ergonomics (Miller et al., 2009).

The index and little fingers can also be abducted. Corresponding positions are defined by rotating the relaxed position 15 ° about the joint between the finger and the palm, which has been assessed to be a comfortable angle for abduction (Lin et al., 2000). In the next sections, these positions will be called *relaxed*, *left*, *right*, and *down*, according to the positions depicted in figure 44.

Using Baudel et al.'s (1993) terminology again, we refer to the chords defined from these additional positions as "*tense*" chords, since they require additional muscle tension in some fingers.

If we consider all the possible positions for each finger, and the fact that it can be lifted, the second guideline leads us to a total of 5 x 4 x 3 x 3 x 4 = 720 possible chords. By removing all the tense chords based on the eight finger combinations that are excluded by the first guideline, we are left with 480 chords[4] following the two guidelines. We expect this *expressive* vocabulary to be *operable* and to minimize discomfort.

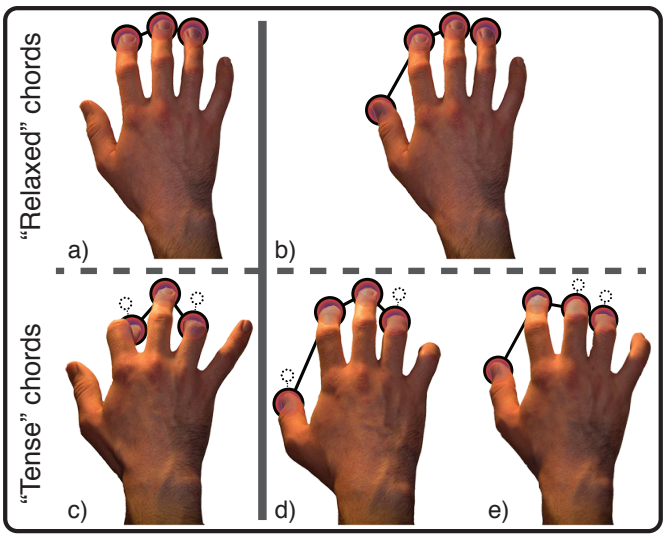


Figure 45: Sample chord set respecting the two guidelines.

Figure 45 shows a sample chord set taken from this vocabulary: *(a)* and *(b)* only include fingers in their *relaxed* positions; *(c)*, *(d)* and *(e)* take advantage of the degrees of freedom of the hand. Although our two guidelines are designed to extend the vocabulary of multi-finger chords while minimizing discomfort, their relevance should be validated empirically.

5.3 EXPERIMENT 1: ASSESSMENT OF CHORDS DESIGN

This experiment focuses on novice users' subjective evaluation of chords. The goal of this experiment is to assess the validity of our guidelines by collecting qualitative assessments from potential novice users. Participants are asked to perform a relevant set of chords and,

4. From the finger combinations presented in figure 43, six tense chords can be created from the second combination, six from the sixth, 18 from the seventh, 18 from the ninth, 24 from the 16th, 24 from the 21st, 72 from the 22nd and 72 from the 24th. All these 240 tense chords might be uncomfortable according to our first guideline.

for each chord, to assess if it is *understandable* and *comfortable* to perform. The experiment has three phases: two evaluation phases A and C, separated by a practice phase B with no subjective evaluation in order to study if the participants' perception of comfort and understandability change after acquiring some experience and having a clearer overview of the vocabulary.

5.3.1 *Chord Vocabulary*

In order to explore the design space of multi-finger chords based on our guidelines, we test a vocabulary of 52 chords: the 26 relaxed chords (see Fig. 43) + 26 tense chords. In order to validate the relevance of our first guideline, we include the eight relaxed chords that are supposed to be uncomfortable.

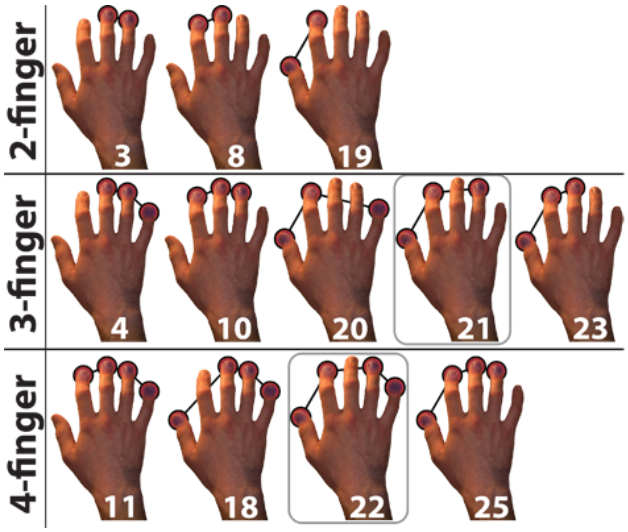


Figure 46: The "relaxed" chords which have been preferred in our pilot study.

For the second guideline, we created a representative set of 26 tense chords: three 2-finger chords, ten 3-finger chords, eight 4-finger chords, and five 5-finger chords. We intentionally considered more chords with fewer fingers, since they should be more *accessible* to novices and yet provide a high level of *expressiveness*. Nevertheless, we featured fewer 2-finger chords since they are less representative of chording interaction.

We first conducted an informal pilot with 12 participants to study users' preference among relaxed chords only, and generated tense chords from the preferred ones. The apparatus, design and task were the same as in the experiment. We used the three 2-finger chords, five 3-finger chords and four 4-finger chords from the relaxed vocabulary that received the best ratings (see Fig. 46). Each 3-finger and 4-finger chord from this selection was used to generate two tense chords, while each 2-finger and 5-finger chord was used to generate one tense chord.

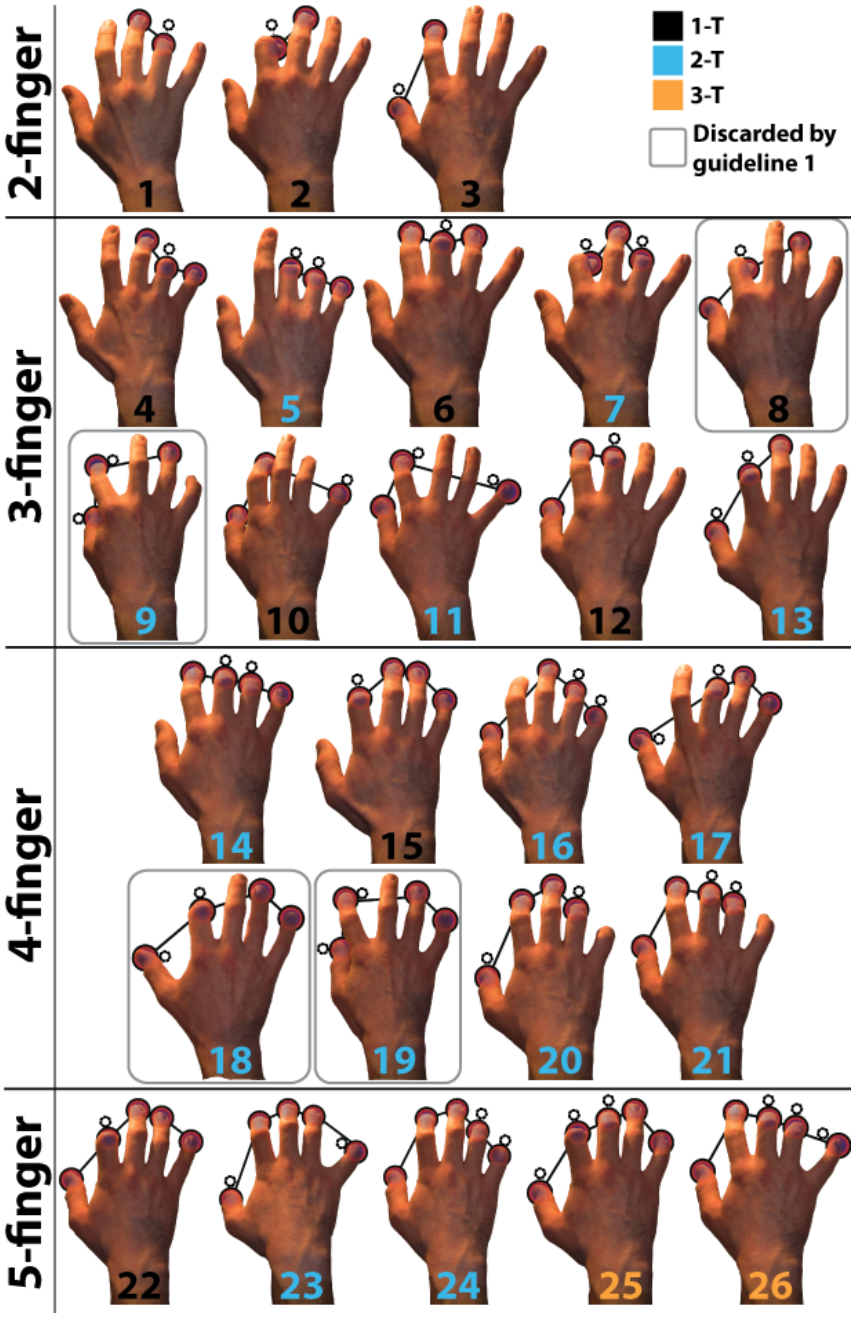


Figure 47: The 26 "tense chords" (CT01 – CT26) used in this experiment, created from the 13 preferred "relaxed" chords in our pilot study.

Among the chords involving the same number of fingers, we balanced the number of fingers in tense positions, which will be abbreviated as follows: *1-T* chords have one finger in a tense position, *2-T* have two, and so on. Our tense vocabulary is finally made of: three *1-T* 2-finger chords; five *1-T* and five *2-T* 3-finger chords; one *1-T* and seven *2-T* 4-finger chords; one *1-T*, two *2-T* and two *3-T* 5-finger chords (see Fig. 47).

5.3.2 *Hypotheses*

We have five hypotheses about the subjective evaluation of this 52-chord vocabulary: (i) ratings will increase between the two testing phases; (ii) ratings will be higher for relaxed chords than for tense chords; (iii) ratings will drop when the number of fingers increases; (iv) for tense chords, ratings will drop when the number of fingers in tense positions increases; (v) ratings on comfort will be lower when the first guideline is not respected.

5.3.3 *Apparatus & Participants*

The experiment was implemented in Java and conducted on an Apple MacBook Pro with an external 3M 27" multitouch screen installed horizontally on a table (95 cm height). We recruited 12 unpaid volunteers (two female) who had not participated in the pilot, all right-handed, their age ranging from 21 to 39 (mean 28, median 27). Two of them had no prior experience using multitouch devices, two were casual users, the others used multitouch smartphones or tablets every day.

5.3.4 *Task & Stimulus*

During the evaluation phases A and C, a trial consists in reproducing a chord and marking perceived *understandability* and *comfort of use*.

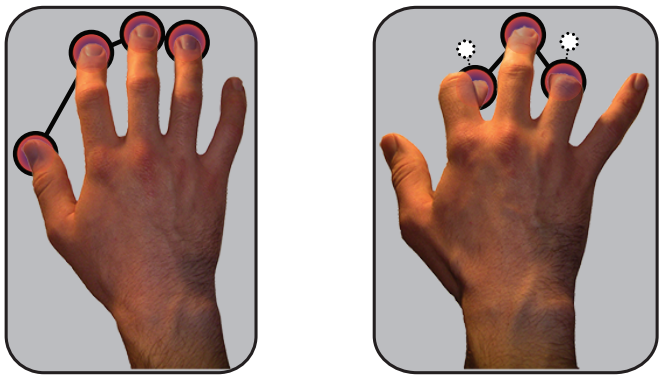


Figure 48: The stimulus for relaxed and tense chords in this experiment.

First, a picture of a hand performing the chord is presented (see Fig. 48), with colored circles drawn under the fingers that must touch the surface, as in common cheat sheets, e.g., in FingerWork's manuals (FingerWorks, 2001). We adapted cheat sheets to our *expressive* vocabulary of chording gestures: if a finger is in a tense position, a smaller dashed circle indicates its relaxed position as a reference to improve participants' awareness of the possible finger positions.

When the stimulus appears, the user has to perform the chord at least twice. A visual feedback similar to the stimulus is provided

during performance, made of linked red dots under detected finger contacts. Then, the two statements we use for subjective evaluation are displayed: *"I find this chord easy to understand and reproduce"*; and *"I find this chord comfortable to perform"*. The users select their answer on a five-level Likert scale: *strongly disagree*, *disagree*, *neither*, *agree* and *strongly agree*. After giving their ratings, participants press a button to go to proceed with the next chord.

During the practice phase B, a trial only consists in the reproduction task without assessment. In that phase, our chord recognizer (described later in the next section) returns a "success" / "error" feedback after each trial to encourage participants to carefully reproduce the chords.

5.3.5 *Design & Procedure*

The experiment is a within-subject design with CHORD (CR01 – CR26 & CT01 – CT26) as the only factor. At the beginning of a session, three chords are performed by the experimenter to demonstrate the reproduction task. Participants are instructed to calibrate our recognizer by maintaining their hand in a relaxed position on the surface (details about the recognizer are reported in the next section). The task and procedure are then presented, and the participants are instructed that the evaluation phases are the core of the experiment, while the goal of the practice phase is to rehearse. In a warm-up session, each chord is presented once to the participant and reproduced with no evaluation nor feedback on recognition. This ensures that the participants get familiar with the stimulus and the device, and get an overview of the whole vocabulary before the first evaluation phase.

Each chord is presented once during each evaluation phase, and three times during the interleaved practice phase. Within each phase, all chords are in random order and organized into blocks of seven, allowing for short breaks between blocks. We collected 104 evaluation trials (user's assessment of *understandability* and *comfort* in phases A & C) and 156 practice trials (chord recognition *success* in phase B) for each participant.

Participants are asked to perform the experiment with their dominant hand, and to perform the chords in the middle of the multitouch screen. An experimenter controls the warm-up session and the first evaluation phase to ensure that participants are able to reproduce all the chords before giving their ratings. A session lasts about 40 minutes after which participants are given a questionnaire to evaluate the task and the recognizer feedback on 5-point Likert scales.

5.3.6 *Understandability and Comfort of Chords*

Our two only measures are UNDERSTANDABILITY and COMFORT, which are the ratings given by the users on the Likert scales in the testing phases A and C. We analyze the results according to CHORD TYPE, TESTING PHASES, NUMBER OF FINGERS, NUMBER OF "TENSE FINGERS" and GUIDELINE, which indicates whether or not a chord follows our first guideline.

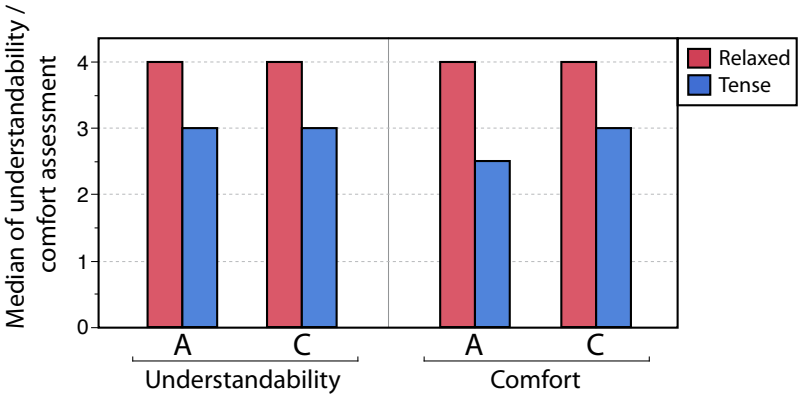


Figure 49: Comfort and understandability by testing phase (A and C)

We first study[5] the influence of TESTING PHASES and CHORD TYPE on the subjective evaluations provided by the participants. These results are presented in Figure 49[6]. Wilcoxon tests for TESTING PHASES —i.e. phases A and C— reveals a significant effect on COMFORT ($\chi^2(1) = 4,3787$, $p = 0,0364$) but no significant effect on UNDERSTANDABILITY ($\chi^2(1) = 3,1768$, $p = 0,0747$). As shown in Figure 49, this effect on COMFORT seems to be related to *tense* chords. In fact, a Wilcoxon test for TESTING PHASES considering only *tense* chords reveals a significant effect on COMFORT ($\chi^2(1) = 4,9397$, $p = 0,0262$), while for TESTING PHASES considering only *relaxed* chords, it does not reveal a significant effect on COMFORT ($\chi^2(1) = 0,7663$, $p = 0,3814$). These results suggest that the participants do not perceive chords as more *understandable* after a short practice session —phase B—, but they feel more comfortable with performing *tense* chords over time. However, as we can see in figure 49, the difference between the median values is small in each case. The effect of the testing phase is negligible on both of our measures. Therefore, hypothesis (i) —i.e. "ratings will increase between the two testing phases"— is not verified, and in the rest of the analysis, we analyze the participants' assessment without distinguishing between the phases.

A Wilcoxon test for CHORD TYPE —i.e. *relaxed* or *tense*— reveals a significant effect on both COMFORT ($\chi^2(1) = 282,3564$, $p < 0.0001$)

5. Analysis were performed with the SAS JMP Pro platform: www.jmp.com/software/pro/

6. As we analyze Likert scales, all figures represent medians.

and UNDERSTANDABILITY ($\chi^2(1) = 287,2755$, $p < 0.0001$) (see Fig. 49). These results suggest that *relaxed* chords are perceived as more *understandable* and *comfortable* than *tense* chords. But overall, median values of the assessments are at least three —i.e. the *agree* item on the Likert scale— for all conditions, except for the comfort of use with *tense* chords in the first testing phase. This suggests that the participants perceive our *expressive* vocabularies of chording gestures as sufficiently *comfortable* and *understandable* for practical use.

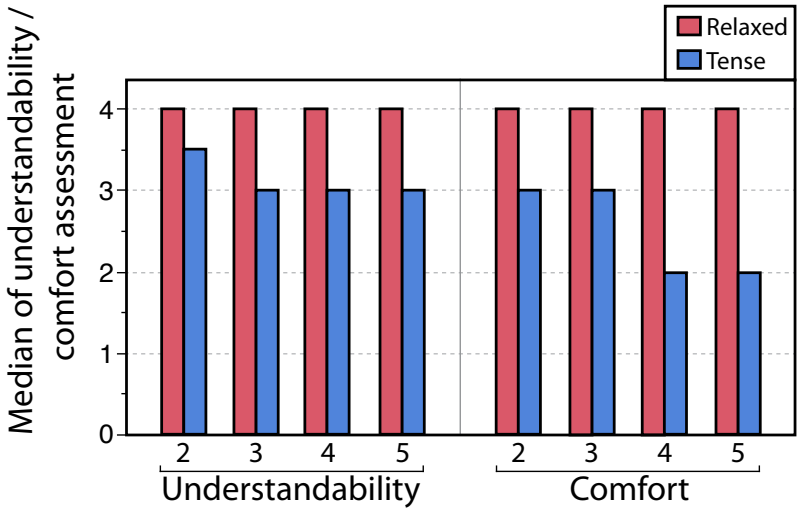


Figure 50: Comfort and understandability by number of fingers

We then study the effect of the number of fingers involved in a chord (see Fig. 50). Kruskal-Wallis tests for NUMBER OF FINGERS reveal a significant effect on UNDERSTANDABILITY for both *relaxed* ($\chi^2(3) = 14,6455$, $p = 0,0021$) and *tense* ($\chi^2(3) = 26,9625$, $p < 0.0001$) chords, as well as a significant effect on COMFORT for both *relaxed* ($\chi^2(3) = 23,1179$, $p < 0.0001$) and *tense* ($\chi^2(3) = 74,5398$, $p < 0.0001$) chords. These results suggest that the number of fingers involved in a chord has an important influence on the *understandability* and *operability* of chording gestures. Figure 50 shows that this features impacts *tense* chords more than *relaxed* chords. Furthermore, we observe that the *comfort* of *tense* chords has a median value below 3 —i.e. the *agree* item on the Likert scale.

For the tense chords, we also analyze the influence of the number of fingers that are in tense positions (see Fig. 51). A Kruskal-Wallis test for NUMBER OF "TENSE FINGERS" reveals significant effects on both UNDERSTANDABILITY ($\chi^2(2) = 65,6838$, $p < 0.0001$) and COMFORT ($\chi^2(2) = 147,2311$, $p < 0.0001$). As we see in figure 51(a)), this effect seems more important for *comfort* than for *understandability*.

Finally, we verified our first guideline (see Fig. 51(b)). Wilcoxon tests for GUIDELINE reveal a significant effect on COMFORT for *relaxed* chords ($\chi^2(1) = 105,4548$, $p < 0.0001$), but not for *tense* chords

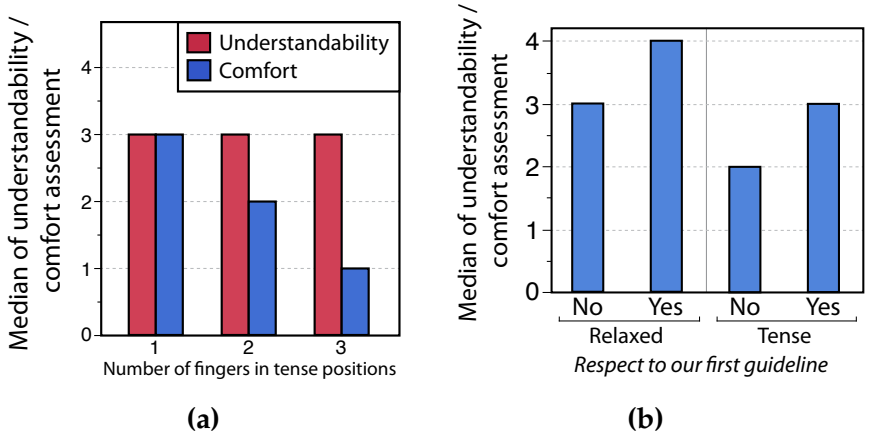


Figure 51: (a): Tense chords: Comfort and understandability by number of fingers in tense positions; (b): Comfort by agreement with our first guideline

($\chi^2(1) = 3,8162$, $p = 0,0508$). Therefore, the influence of features considered by our first guideline seems stronger for *relaxed* chords. For *tense* chords, the number of involved fingers and the number of fingers in tense positions seem to override this effect.

5.3.7 *Post-experiment questionnaire*

In the post-experiment questionnaire, all participants pointed out the difficulty of understanding precisely where to put the fingers from a static image displayed at a distance from the hand, although this consideration was not observable in our statistical analysis. However, all of them reported that they found *tense* chords to be more difficult to *understand* and to *perform*.

Four participants found 2-finger and 5-finger chords easier to perform than 3-finger and 4-finger chords. For these two latter kinds of chords, they had more difficulties in understanding which finger was involved and which was the appropriate position for each of them. Three of the participants observed that chords from which they can extract geometric features (e.g. alignment) were easier to perform (see for example the alignment of finger positions in CT22 in Fig. 47).

5.3.8 *Summary And Discussion*

According to these results:

- Hypothesis (i) —i.e. "ratings will increase between the two testing phases"— is not verified;
- Hypothesis (ii) —i.e. "ratings will be higher for relaxed chords than for tense chords" — is supported;

- Hypothesis (iii) —i.e. "ratings will drop when the number of fingers increases"— is supported but the impact of this feature seems more important for *tense* chords;
- Hypothesis (iv) —i.e. "for tense chords, ratings will drop when the number of fingers in tense positions increases" — is supported, but the impact of this feature seems more important regarding *comfort* than regarding *understandability*;
- Hypothesis (v) —i.e. "ratings on comfort will be lower when the first guideline is not respected" — is supported for *relaxed* chords, but not for *tense* chords.

Overall, the participants gave high ratings to both the *understandability* and *comfort* of our *expressive* chord vocabulary, which promotes its use in applications. However, the results suggest that the *understanding* of the most complex chords —i.e. *tense* chords involving at least three fingers— may require support.

As we want to maximize the *understandability* of *expressive* chord vocabularies, our next challenge is to create a simple and effective learning method for chording gestures, inspired by novices' spontaneous strategies to tackle the complexity of *tense* chords.

5.4 ARPÈGE: A DYNAMIC GUIDE FOR CHORDING GESTURES

According to related work and to our observations while conducting the first experiment, we identified the three following requirements for our learning system.

ONLINE GUIDANCE: Dynamic guides, such as OctoPocus (Bau and Mackay, 2008), have strong advantage for gesture learning: they do not require to leave the application context; they occupy a moderate amount of screen real estate since the whole vocabulary is reachable with just one guide; they progressively teach the user *how* to properly perform the gesture while doing it, in addition to showing *what* the gesture is. Therefore, they help make *expressive* gesture vocabularies *understandable* and *operable*. Our first goal in this section is to design such a guide adapted to chording gestures.

GRAPHICAL OCCLUSION AND VISUAL COMPLEXITY: Our learning system must show whether a finger is part of a chord or not, as well as its appropriate position among those we have defined: relaxed, right, left or down. However, it should not directly display all the information about every chord in the vocabulary, in order to minimize visual complexity, as is the case with OctoPocus (Bau and Mackay, 2008). Furthermore, we must address the common issue with direct tactile interaction: graphical occlusions. For example, Vogel and Casiez created "occlusion shape models" to help designers

avoid creating tactile user interfaces where visual information is occluded by the hand or the forearm of the user (Vogel and Casiez, 2012).

CHORDS BREAKDOWN: Although participants in our first experiment felt able to reproduce most of the relaxed chords from the stimulus, many of them reported difficulties with preparing their hand before putting their fingertips on the surface, especially for tense chords. We observed that they spontaneously developed two main strategies to overcome chords complexity:

- putting down fingers on the surface one after the other (7 participants);
- putting fingers in the relaxed positions first before moving them to tense positions if needed (2 participants).

The first strategy reduces chords complexity by sequentially focusing on each finger. The second one splits the complex movement of reaching a tense position into two simpler moves. We base our dynamic guide on these strategies since they proved efficient in the expert activity of playing music. In fact, novice musicians often face similar problems related to *understandability* and *operability* when they sight-read chords from a score. To address these problems, music teachers encourage students to first play an arpeggio, pressing the keys one after the other, to experience a step-by-step understanding of a chord; and to consider the most complex chords as alterations or diminutions of basic ones.

5.4.1 *Design*

With these requirements in mind, we designed *Arpège*, a dynamic guide for chorded command triggering. Similarly to Octopocus (2008), *Arpège* combines feedforward and feedback to provide progressive guidance to novice users. Feedforward shows which chords remain feasible and what actions are required to complete them. Feedback indicates which finger positions have been identified thus far, and if an entire chord has already been recognized.

Before describing the design of *Arpège* in more detail, we illustrate its use through a simple scenario (see Fig. 52).

USING ARPÈGE: Mireille wants to learn how to use chords to trigger the *copy* command in her favorite multitouch drawing application. She wants to keep her focus on her drawings and get rid of distant menus, but she is unfamiliar with chords. To invoke *Arpège*, she quickly taps the surface with the five fingertips of one hand, and removes them from the surface. As shown in Figure 52(a), *Arpège* displays groups of fingerprints representing all possible positions for every finger, along with labels indicating the first finger to lay on

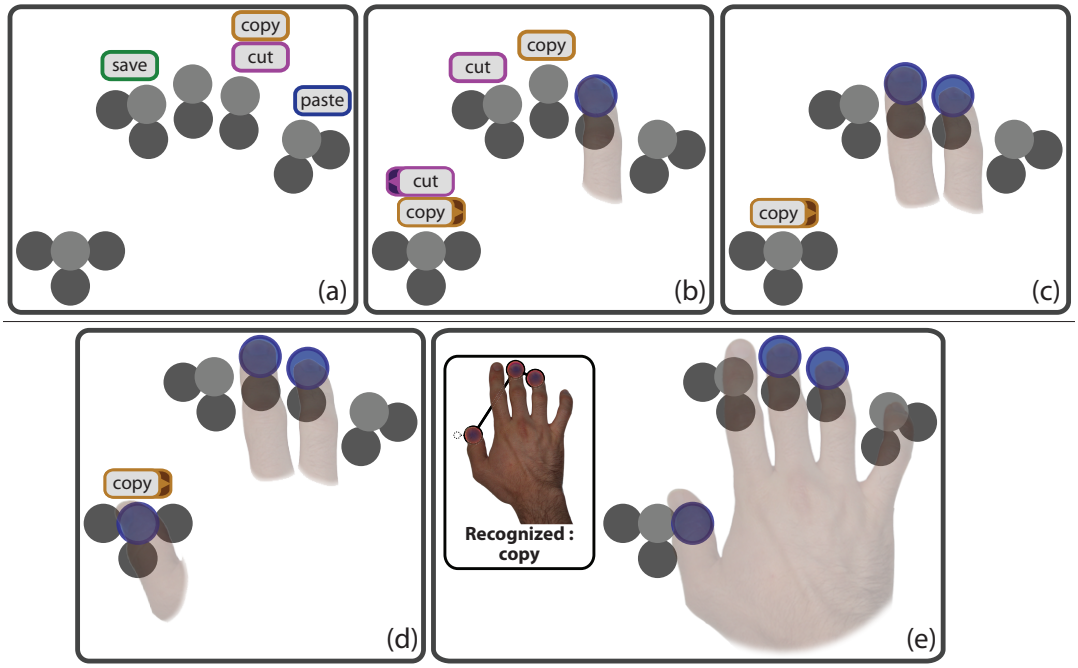


Figure 52: Triggering the *copy* command with *Arpège*. (a) Invoking *Arpège* displays fingerprints representing possible finger positions as well as command labels. (b) Touching the fingerprint under the *copy* label with the ring finger discards *paste* and *save* since they are no longer reachable. (c) When the middle finger is put down on the appropriate fingerprint, the *copy* label disappears. (d) If the thumb is laid on the fingerprint corresponding to the relaxed position, the label does not disappear, indicating that the thumb must be moved to the right. (e) The thumb reaches the *right* position, the chord is complete and the user is notified that *copy* will be triggered if fingers are lifted together in the current configuration.

the surface for each command. Since she wants to trigger the *copy* command, Mireille places her ring finger on the corresponding fingerprint (see Fig. 52(b)). As a result, *Arpège* displays new labels showing what other actions are required to perform any chord starting with the ring finger, and discards those which can no longer be performed (e.g., the *paste* command starting with the little finger).

Mireille continues along the *copy* path, placing her middle finger (see Fig. 52(c)) and her thumb (see Fig. 52(d)) on the indicated fingerprints which are highlighted when touched. Note that the *cut* command disappeared since it can no longer be recognized. But if Mireille lifts her middle finger and thumb, *Arpège* will return to the previous state and make the *cut* command reachable again. Although Mireille has put the three appropriate fingers on the surface, one last *copy* label remains visible, indicating that there is still a movement to perform: the right arrow next to the label instructs Mireille to move her thumb to the right. When she moves her thumb to that position,

the *copy* chord is recognized (see Fig. 52(e)). A picture of a hand performing the corresponding chord gives a summary of the movements that have been done following *Arpège*. Finally, when Mireille lifts her fingers simultaneously in that configuration, *Arpège* disappears and the *copy* command is triggered.

BEFORE USING ARPÈGE: For each new user, *Arpège* requires a short calibration process to adapt the dynamic guide and the chord recognizer to the user's hand shape and size. In our implementation, the user puts her fingertips on top of five on-screen circles to register the fingers (see Fig. 53). Then, she moves her fingers to reach the most comfortable hand position with all fingertips in contact with the surface and the palm roughly parallel to it. Finally, with a finger of her other hand, she inputs the positions of the two joints that are respectively between the index finger and the palm, and between the little finger and the palm (small black dots on figure 44). From these seven points, the system is able to infer all the possible finger positions defined by our guidelines.

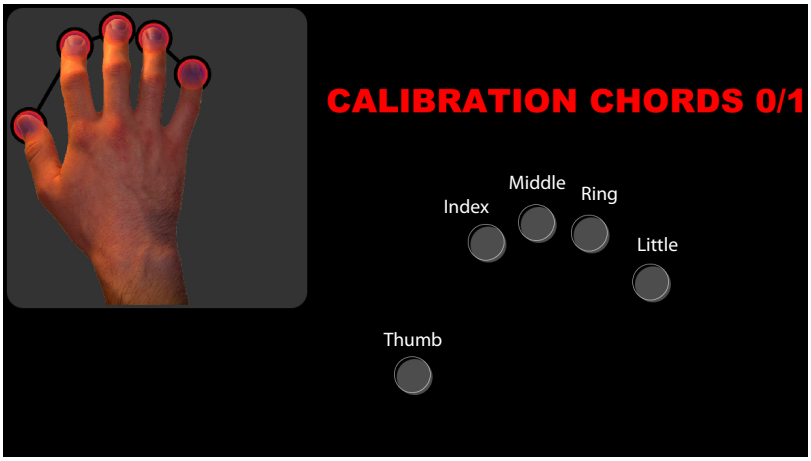


Figure 53: The calibration process of *Arpège*.

Arpège can be invoked in several ways. In our scenario, Mireille performed the 5-finger relaxed chord, but it can also be called via an on-screen or physical button, or by performing a double tap. Another possible and non-modal method to invoke the guide, similar to Marking Menus (Kurtenbach, 1993) and Octopocus (Bau and Mackay, 2008), would be to automatically trigger *Arpège* when the user hesitates while performing a chord (i.e., not moving for a certain amount of time). However, a finger identification method would be needed in this case for *Arpège* to identify which fingers are touching the surface, and thus what guidance must be provided. We did not explore this solution since, to our knowledge, all the finger identification methods in the literature require an additional camera —e.g. Kung et al.

(2012)—, or specific kinds of multitouch devices —e.g. Lepinski et al. (2010).

VISUAL LAYOUT: *Arpège* is made of two dynamic graphical layers: fingerprints and command labels. Fingerprints are displayed as circles which locations depend on the calibration. Tense positions are darker than relaxed ones. From the user's point of view, these targets show comfortable positions. From the system's point of view, they allow finger identification.

Since it is up to the user to place the right finger on the corresponding circle, the system can be faulted in case of a mistake (e.g., touching the ring finger circles with the little finger). However, we never observed such situations during the experiment, which suggests that these visual clues convey enough information for the users to understand what are the appropriate positions for every finger.

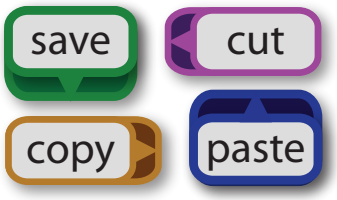


Figure 54: *Arpège*'s cartouches and labels.

Labels are made of colored command names inside cartouches indicating the actions to perform for each command. Colors are associated with commands to ease the visual search. Cartouches and labels (see Fig. 54) combine information about which finger to put on the surface (depending on the group of fingerprints above which they are located) and where to put it (thanks to the arrows on the side). When a finger is involved in several chords, the labels of the corresponding commands are stacked on top of its fingerprints (see Fig. 52(a)&(b)). Stacked cartouches are ordered consistently amongst fingers, in an order that can be changed dynamically (e.g., alphabetically, most used command, etc.). Labels and arrows remain displayed until the position they indicate is reached by the user. Therefore each cartouche represents an action that remains to be performed.

ARPÈGE AND OUR REQUIREMENTS: *Arpège* provides progressive guidance for an entire vocabulary in the context of the application, and thus does not require as much screen space as cheat sheets or videos.

Since the feedback when touching a fingerprint is a circle larger than the fingertip, it can be seen from any side of the finger. Placing labels above the fingerprints is in line with the guidelines provided by Vogel et al. (2012) to avoid occlusion on multitouch displays. We also limit occlusions by encouraging users to begin gestures with

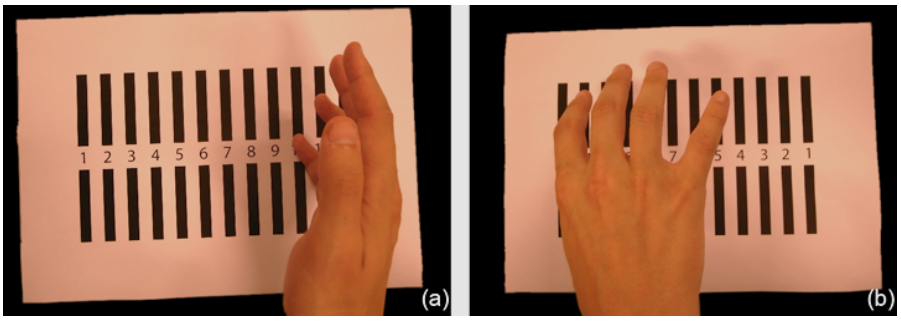


Figure 55: a) Placing the *outmost* finger (the furthest finger from the thumb) first reduces occlusion. b) Placing the thumb first increases occlusion. Both thumb and little finger are placed on the vertical segment #10 on these pictures.

the "outmost" finger first (see Fig. 55). Visual complexity is limited by indicating only the starting position for each chord in the initial state (see Fig. 52(a)), and by displaying further guidance only for the chords that are still reachable (see Fig. 52(b)&(c)). When an existing chord is recognized by the system during the interaction, no more corresponding labels are displayed since all the positions have been reached. *Arpège* also displays a picture of this chord as an additional feedback and feedforward.

Since the fingers are at proper positions only after following *Arpège*'s guidance, the command must not be triggered before the user lifts all her fingers simultaneously in the appropriate positions. If a finger is lifted while *Arpège* is being used, the system goes back to the previous state, allowing the user to correct an incorrect finger placement. The user can thus explore the entire vocabulary and discover the available commands by laying, moving and lifting fingers. If the user lifts all her fingers in sequence, no command is triggered.

As is the case with Marking Menus (Kurtenbach, 1993) or Octopocus (Bau and Mackay, 2008) for gesture-marks, the immediate chord gestures performed by experts are identical yet faster than novices' guided chords. This similarity should allow a fluid transition from novice to expert (Scarr et al., 2011). We also observed in the experiments that some users may have a partial knowledge of chords without being experts. They used *Arpège* to remind them how to start the chord: after laying the first finger, one can directly put all the remaining fingers at the same time. This possibility also helps the transition from novice to expert, as reported in OctoPocus (Bau and Mackay, 2008), where intermediate users call the dynamic guide but do not need more information than how the gesture starts to remember how to perform it. We also believe that the picture of the chord displayed when a chord is recognized should help users understand and memorize the vocabulary, as with ShadowGuides (Freeman et al., 2009).

5.4.2 *Implementation*

Arpège is made of three software components: (i) the input and output module, in charge of capturing user's touches and displaying feedforward and feedback; (ii) a partial chord recognizer, which infers potential target chords as well as required actions for completion, given the finger positions that have been detected; (iii) a chord classifier, which compares the geometric features of the final chord to the templates in the vocabulary and returns the best match.

Visual feedback is implemented with custom Java2D graphics. The input module uses a TUIO listener[7], in charge of interpreting and grouping user's touches into potential chords, and sending the list of current reached positions to the partial recognizer. Final positions are sent to the chords classifier when all the fingers are lifted within 100 ms.

Chord templates are represented in two ways in the system: (i) an abstract representation, made of the list of involved fingers and their positions (i.e., relaxed, down, left and right), which is used by the partial recognizer; and (ii) a geometrical representation, describing normalized distances and angles between the points of a chord. These features are computed from the user's hand calibration and our definition of tense positions (i.e., 2 cm translations and 15 ° rotations), and are used by the chord classifier.

ARPEGE'S PARTIAL CHORD RECOGNIZER: The partial recognition algorithm provides the step-by-step feedback and feedforward of *Arpège*. For that recognizer, the vocabulary is modeled with a tree. Every node corresponds to a list of positions, and maps to a list of reachable chords and required movements to complete these. Events are sent by the TUIO listener when available positions (grey circles on Fig. 52) are reached or left. These events trigger a tree traversal algorithm which updates its current state to the corresponding node. Information on the current node are then sent to the output module, which in turn updates the dynamic guide and displays the picture of the corresponding chord in case a leaf is reached —i.e. if a chord is recognized.

A CHORD CLASSIFIER: The chord classifier does not depend on the dynamic guide and can be used independently. It recognizes chords from a set of chord templates by comparing geometric features, independently of our definition of relaxed and tense positions.

Our algorithm compares an input chord to the templates in the vocabulary that have the same number of points. It computes the

7. TUIO is a standard communication protocol for touch events based on OSC. See the TUIO website at http://www.tuio.org/

squared point-to-point distance —i.e. the sum of the squared distances between respective positions— between all the permutations of the input chord and each template in the vocabulary. It is scale and rotation tolerant, by searching for a combination of scale and rotation that minimizes the distance between each permutation of the input chord and each template in the vocabulary. Scale is tested between 0.9 and 1.1 times the size of the input chord, and the rotation tolerance ranges from 15° clockwise to 15° counterclockwise. According to our tests, these tolerances allowed efficient chord recognition, and informal observations from participants suggest that it provides additional comfort when performing chords. Then, the minimum distances between each permutation and each template are compared, and the best match among the whole vocabulary is returned.

The classifier is triggered when an expert user performs a chord without invoking *Arpège*. However, to maintain the consistency between novice and expert modes, this classifier also processes the positions sent by the TUIO listener when the user lifts her fingers after following *Arpège*. We ran a benchmark of this classifier with data from the previous experiment: all the chords were recognized by the classifier if they were recognized by the partial recognizer, i.e. if appropriate finger positions were detected.

Unlike ShadowGuides (Freeman et al., 2009) and GesturePlay (Bragdon et al., 2010) which used a *Wizard of Oz* recognition process for the purposes of their experiments, our recognizers make *Arpège* directly usable in real applications. In addition, the whole system is mostly user-independent, since it only requires a very short calibration process, and does not require a user-dependent training set.

With this recognizer, the success rates in the practice phase B of the first experiment are 76% for relaxed chords and 80% for tense chords. It shows that novice users succeeded most of the time in reproducing chording gestures although they were not instructed to focus on the accuracy of reproduction. This recognition rate is still lower than what is usually acceptable for a quality user experience. This indicates that our recognizer should take into account the geometric rules we use for designing chording gestures in order to improve the *operability* of our *expressive* chord vocabularies. Nevertheless, since *efficiency* is not the focus of this study, we did not design an experimental setup allowing to distinguish between errors from the device (e.g., problems in detecting a contact), errors from the participant (e.g., wrong number of fingers, wrong finger positions) and errors from the recognizer. Therefore, a specific study should be conducted in order to identify these errors and improve our analysis of errors, thus improving the external validity of the system.

In the following section, we report on an experiment evaluating the suitability of *Arpège* for learning chording gestures.

5.5 LEARNING AND MEMORIZATION WITH ARPÈGE

In the first experiment, participants reported problems in *understanding* complex chords presented by the cheat sheet. In this second experiment, we compare *Arpège* to a cheat sheet when learning and memorizing chording gestures as command triggers. Since we are not only interested in learning rates and performance, but also in *understandability*, we give users a post-experiment questionnaire to study how both learning methods help them make sense of chording gestures. Similar to the study of the Gesture Play system (Bragdon et al., 2010), we expect *Arpège* to improve memorization since it provides more information than the cheat sheet about *what* gesture to perform, *how* to perform it and *why* it is different from other gestures.

5.5.1 *Hypotheses*

Our hypotheses about the learning and memorization of chording gestures as command triggers are as follows: (i) participants will be able to learn and memorize a representative set of chording gestures; (ii) learning will be faster with *Arpège* than with the cheat sheet; (iii) mid-term memorization will be better with *Arpège* than with the cheat sheet; (iv) participants learning chords with *Arpège* will have more chords recognized by the classifier than participants learning chords with the cheat sheet.

5.5.2 *Apparatus & Participants*

We used the same computer and multitouch screen as in the first experiment. A physical button —Griffin PowerMate USB Multimedia Controller— is used to invoke the help systems. This setup avoids interference between help invocation and chorded interaction, and minimizes contextual differences between techniques when studying memorization. In fact, software cheat sheets are mostly reachable via menus, while dynamic guides are on-line learning methods, making the former even slower. We recruited 24 participants and allocated them to two groups of 12: the first group learned chords with the cheat sheet, the second with *Arpège*. In each group, four participants had participated in the first experiment. Participants in the first group were between 24 and 41 years old (mean 29, median 28, 3 female), and participants in the second group were between 19 and 30 years old (mean 26.7, median 27, 4 female). Before starting the experiment, they were asked to self-evaluate their overall ability to memorize items by giving it a mark between 1 and 5. Mean and median values were respectively 3.63 and 4 for the first group, 3.41 and 4 for the second group. In the first group, 8 participants used multitouch devices everyday, e.g. multitouch smartphones, while the others have no prior experience with multitouch interfaces. One more participant was a

everyday user in the second group, three were novices. All participants were right handed.

5.5.3 *Techniques*

For the cheat sheet, we used the same visual representation of chords as the stimulus of the first experiment. Like those from FingerWorks (FingerWorks, 2001), our cheat sheet displays all available chords in a table. It also provides additional information on relative finger positions in line with the vocabulary we designed, such as the position of relaxed positions as a reference for tense positions (see figure 48). The picture superimposed on the graphical representation of chord points shows a proper hand posture for comfortable execution.

5.5.4 *Vocabulary*

In order to evaluate chord learning and memorization, we created a representative subset of the 52-chord vocabulary tested in the first experiment. To inform this choice, we modeled chord difficulty using simple rules inspired by the literature on hand mechanics and our observations from the first experiment, by computing a sum of "penalties":

- Chords with three and four fingers have been reported as being the most difficult to perform, so 3-finger chords receive one penalty, and 4-finger chords two penalties since more information must be understood and memorized;
- From the first experiment and from studies of the hand, middle and ring fingers are the hardest to move without disturbing the positions of the other fingers. Therefore, chords where either the middle or ring fingers is lifted along with at least one of the neighboring fingers receive one penalty, while the chords where one of these fingers is lifted while both its neighboring fingers are on the surface receive two penalties (according to the first guideline, presented in section 5.2.1);
- Finally, Participants reported tense chords to be more difficult as the number of fingers in tense positions increases, and this observation has been validated by our analysis. Thus, a chord receives one penalty for each finger in a tense position.

Given these rules, the 52 chords from the first experiment have a total of zero to six penalties.

We then classified the 52 chords of this vocabulary. Chords with a score of zero to two penalties are supposed to be "easy" to perform, while chords with three or four penalties are "medium", and chords with more than four penalties are "hard". Within this classification, 20 relaxed chords are considered easy and the other six are medium.

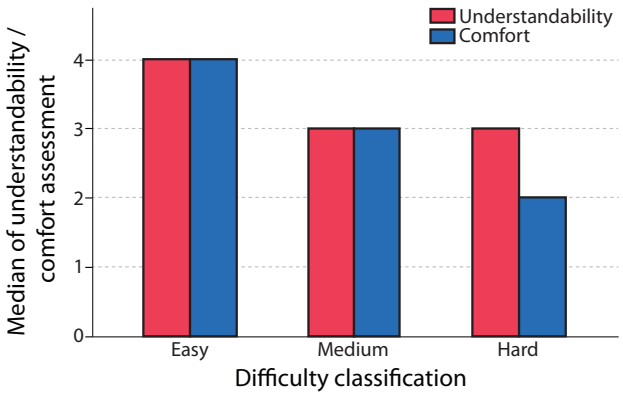


Figure 56: Users' ratings for *understandability* and comfort of use from the first experiment by difficulty classification

For tense chords, five are "easy", 14 "medium" and seven "hard". Figure 56 shows that this classification in difficulty levels is consistent with the users' assessments in the first experiment.

For this experiment, we create a representative chord set by taking the four chords with the best subjective marks for both *understandability* and comfort in each of these categories (see Fig. 57).

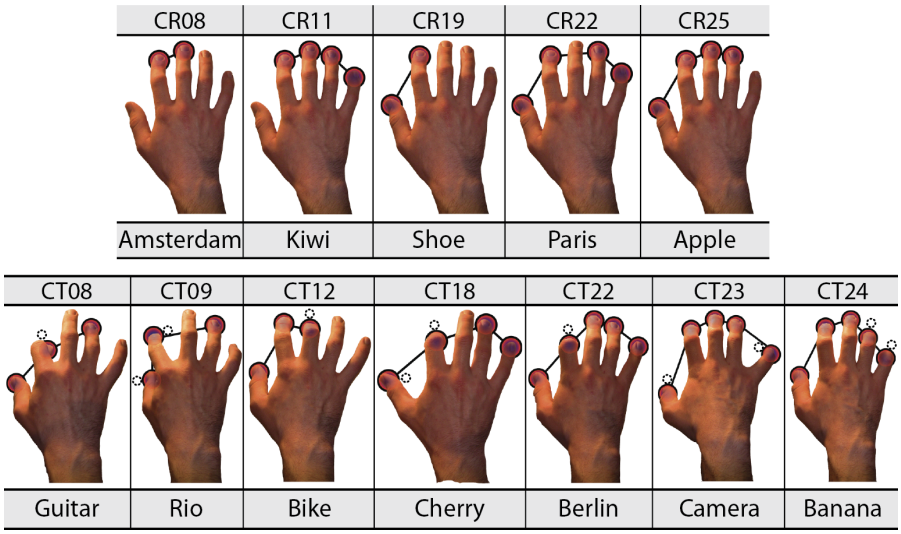


Figure 57: The representative chord set and mapping to commands used in this experiment. CRxx and CTxx refer to the names of the chords in the first experiment (see Figure 43 and Figure 47).

The same vocabulary was used for both groups of participants. Twelve command names from three lexical categories were randomly assigned to the chords: cities (*Amsterdam*, *Berlin*, *Paris*, *Rio*), fruits (*Apple*, *Banana*, *Cherry*, *Kiwi*) and objects (*Bike*, *Camera*, *Guitar*, *Shoe*) (see the chord to command mapping in Figure 57).

5.5.5 *Task & Stimulus*

The task consists in triggering a command (Cmd_i) when its name (N_i) is presented, by performing the appropriate chord (C_i). The experiment has two phases: *learning* and *testing*. We use the experimental design presented in chapter four to provide a supervised learning phase in the beginning of the session, test memorization with one of the two TECH techniques available for help, and test mid-term memorization. While *learning*, the user sees one command name, and discovers the corresponding chord with the learning method of his group. Performing the chord taught by the learning method automatically leads him to the next chord. At this point, both learning methods present only the chord that is asked to the participant: only one picture is presented by the cheat sheet, and *Arpège* displays the labels of only one command.

In the *testing* phase, participants are presented with the name of the command (N_i) only. They have to remember and perform the corresponding chord. If they do not remember it, they can invoke help by pushing the physical button. In this phase, the learning method of each group displays the whole vocabulary: the cheat sheet displays all chords in a table, and *Arpège* displays the guidance for all commands. The participant searches for the chord corresponding to the command he is asked for, understands how to perform it, performs it, receives a "success"/"error" feedback from the recognizer, and is presented the next chord.

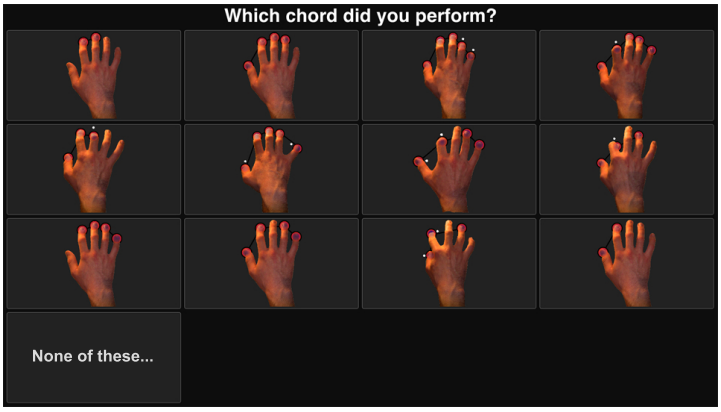


Figure 58: The confirmation system presented after performing a chord without invoking help

If the user remembers the chord corresponding to the command he is presented, he performs it without invoking help. We use the same confirmation system as in the second experiment presented in chapter four, in order to distinguish memorization errors from performance errors: the participant is presented pictures of every chord of the vocabulary and is asked to tap the one he actually tried to perform (see Fig. 58). If the chord he performed is not part of the vocabulary,

he is asked to tap a specific button to indicate he memorized a wrong chord (bottom left button in Fig. 58, labeled "None of these"). After the confirmation, a feedback is provided to tell the user if the chord he memorized, i.e. the one he confirmed by tapping its picture, is the good one and if he performed it well.

Before starting the session, the design is presented to the participants, and they are told to perform a chord in the *testing* phase only if they do not have any doubt on what they have memorized. Otherwise they are encouraged to invoke help. They are also asked to exclusively select the chords that they actually remember for the confirmation question, even if they realize they did not perform the appropriate chord when seeing the entire vocabulary. The experimenter controls the *learning* phase and the first *testing* block to make sure the confirmation mechanism is well understood by the participants. This also allows to make sure the participants do not try to make random chords and cheat when answering the confirmation question.

5.5.6 *Design & Procedure*

We compare two techniques for learning chords (TECH factor): CheatSheet and Arpège. We use a between-subjects design in which participants are randomly assigned to one of the two techniques. Each command Cmd_i is the combination of a command name N_i used as stimulus, and the corresponding chord C_i. The chord to command mapping has been established randomly, and is identical for all participants (see Fig. 57). The command set (CMD factor) has 12 commands: $C_1, \ldots, C_{12}$.

The procedure of this experiment is similar to the procedure of the second experiment presented in chapter four: participants ran through two sessions held on two consecutive days, in order to evaluate learning and mid-term memorization with the two learning methods. On the first day, all participants are presented with a five minutes warm-up session based on a reproduction task identical to the one of the first experiment, to get familiar with performing chords. Then, during the *learning* phase, users are presented each chord twice in random order, and occurrences are grouped in two blocks of 12 trials.

In the *testing phase*, we assigned apparition frequencies following a Zipf distribution (Grossman et al., 2007; Appert and Zhai, 2009) to each of the 12 commands *testing phase*: (13,13,6,6,4,4,3,3,2,2,2,2). As in the second experiment presented in chapter four, this simulates the different frequencies of use of commands in real applications. Frequency assignment is counterbalanced across participants, resulting in the same number of trials for each command overall.

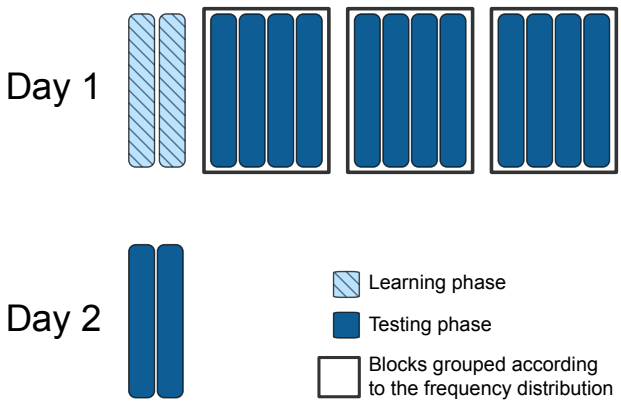


Figure 59: A sample session. Hatched sub-blocks are learning trials, the others are testing trials, grouped according to the frequency distribution on the first day.

On the first day, the participants are presented the complete set of chords three times according to the frequencies distribution, for a total of 180 *testing* trials grouped in 12 blocks of 15 trials (35 minutes mean session duration for the first group, 41 minutes for the second group). Chords occurrences following the frequencies distribution are randomly distributed in four consecutive blocks (see Fig. 59). On the second day, every participant is presented each chord twice. Trials are randomly ordered and grouped in two 12-trial blocks (five minutes mean duration for the first group, six for the second group).

5.5.7 *Quantitative Results*

Our main measures in the testing phases are (i) *recall rate*, the percentage of correct answers to the confirmation question; (ii) *success rate*, the percentage of chords that have been recognized by the recognizer among those for which the participant did not invoke help; and (iii) *help rate*, the percentage of trials where the participant used help.

Before analyzing mid-term memorization on the second day and studying how participants made sense of the mapping between a command name N_i and the corresponding chords C_i, we first study learning during the first day of the experiment. We analyze[8] the results according to TECH and the sub-sessions of the experiment — i.e. the three groupings of the frequency distribution of the first day, and the two 12-chord blocks of the second day— by considering these measures in the model TECH × SUBSESSION × Rand(PARTICIPANT).

According to our between-subject design, we first analyze our experimental data with a multivariate analysis. We check with sphericity tests whether unadjusted univariate F tests are appropriate,

8. All analyses were performed with the SAS JMP Pro platform: www.jmp.com/software/pro/

and then perform unadjusted univariate F tests [9] (repeated-measures ANOVA), which are reported in this section.

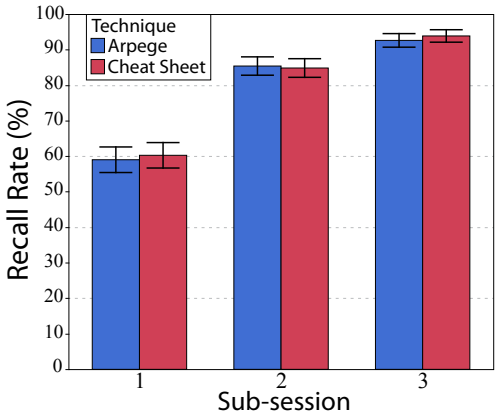


Figure 60: Recall rate for both techniques by sub-session (each error bar is constructed using a 95% confidence interval of the mean).

LEARNING ON THE FIRST DAY: For the first day, we find a significant effect of SUBSESSION on the *recall rate* ($F_{2,44} = 136.0103$, $p < 0.0001$). A Tukey HSD test reveals that the *recall rate* is significantly higher for every successive SUBSESSION (see Figure 60). However, there is no significant effect of TECH on the *recall rate* ($F_{1,22} = 0.0236$, $p = 0.8793$), and no TECH × SUBSESSION interaction effect ($F_{2,44} = 0.1201$, $p = 0.8871$). Figure 60 shows that participants gave appropriate answers to the confirmation question more than 80% of the time in the second sub-session —i.e. after about 10 minutes—, and more than 90% of the time in the third sub-session —i.e. after about 20 minutes.

We find a significant effect of SUBSESSION on the *success rate* ($F_{2,44} = 146.6690$, $p < 0.0001$). As for the *recall rate*, a Tukey HSD test reveals that the *success rate* is significantly higher for every successive SUBSESSION (see Figure 61), but there is no significant effect of TECH on the *success rate* ($F_{1,22} = 0.3012$, $p = 0.5886$), and no TECH × SUBSESSION interaction effect ($F_{2,44} = 0.7212$, $p = 0.4918$). Figure 61 shows that for the trials in which participants did not invoke help within the third sub-session, more than 80% of the chords they performed were recognized by the classifier.

For the *help rate*, an ANOVA reveals a significant effect of SUBSESSION ($F_{2,44} = 98.8849$, $p < 0.0001$), no effect of TECH ($F_{1,22} = 0.0119$, $p = 0.9140$), and no TECH × SUBSESSION interaction effect ($F_{2,44} = 0.3083$, $p = 0.7363$). A Tukey HSD test reveals that the *help rate* is significantly lower for every successive SUBSESSION (see Figure 62). Figure 62 shows

9. Unadjusted univariate F tests can be performed when sphericity tests are validated, according to Davis (2002)

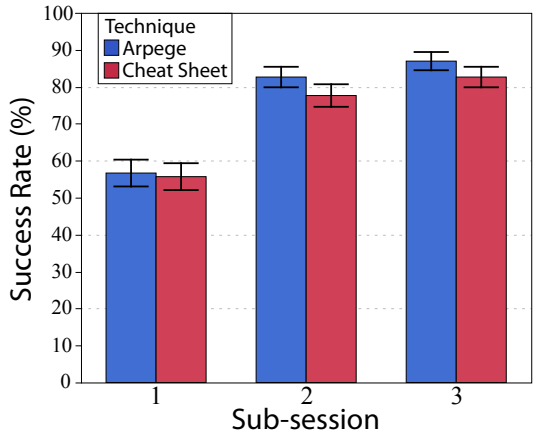


Figure 61: Success rate for both techniques by sub-session (each error bar is constructed using a 95% confidence interval of the mean).

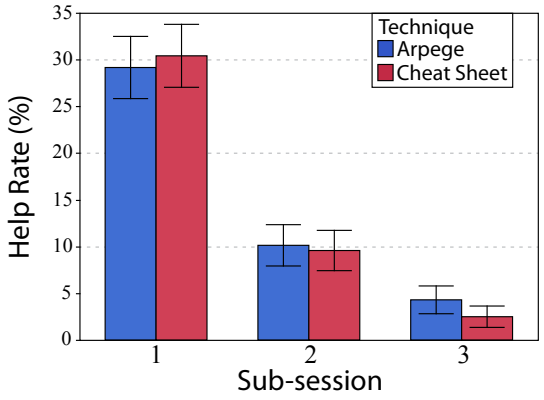


Figure 62: Help rate for both techniques by sub-session (each error bar is constructed using a 95% confidence interval of the mean).

that in the third sub-session —i.e. after approximately 20 minutes—, participants used help less than 5% of the time.

MID-TERM MEMORIZATION ON THE SECOND DAY: An analysis of the *recall rate* over the two days (5 sub-sessions) reveals a significant effect of SUBSESSION on the *recall rate* ($F_{4,88} = 68.9397$, $p < 0.0001$). A Tukey HSD test with Bonferroni correction reveals that the *recall rate* in the first sub-session is significantly different from the *recall rate* in the four following sub-sessions (see Figure 63). This result suggest that memorization on the second day is not significantly different from the high levels of learning reached at the end of the first day. Figure 63 shows that the participants remembered more than 80% of the chords in the first sub-session of the second day. The improvement between the first and the second sub-sessions of the second day might be due to the fact that users remember the differences between chords when they are asked to perform them, and benefit from the summary of the vocabulary provided by the confirmation question.

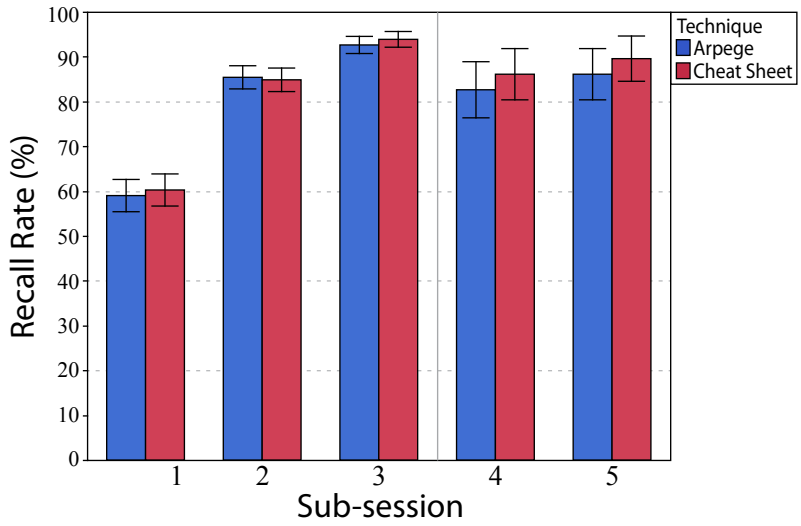


Figure 63: Recall rate over the two days for both techniques by sub-session (each error bar is constructed using a 95% confidence interval of the mean).

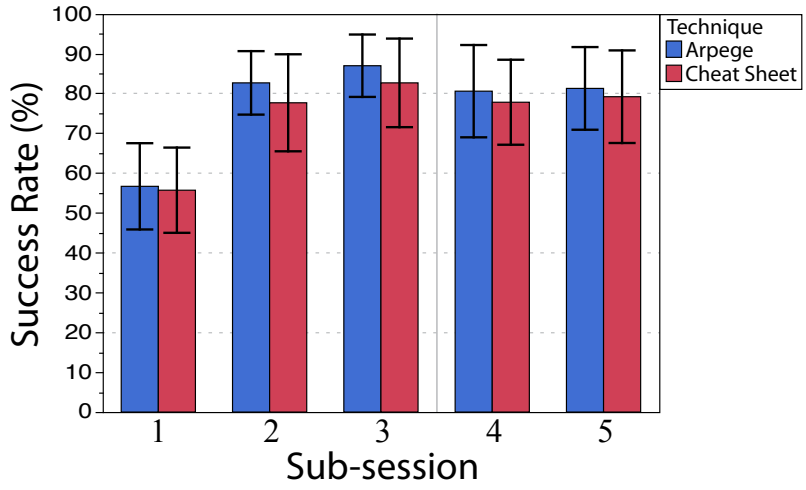


Figure 64: Success rate over the two days for both techniques by sub-session (each error bar is constructed using a 95% confidence interval of the mean)

By analyzing the *success rate* over the two days, we find a significant effect of SubSession on the *success rate* ($F_{4,88} = 56,8545$, $p < 0.0001$). A Tukey HSD test with Bonferroni correction reveals the same result as for the *recall rate* —i.e. the *success rate* in the first sub-session is significantly different from all others—, suggesting that the number of chords recognized by the classifier on the second day is not significantly different from what has been reached at the end of the first day.

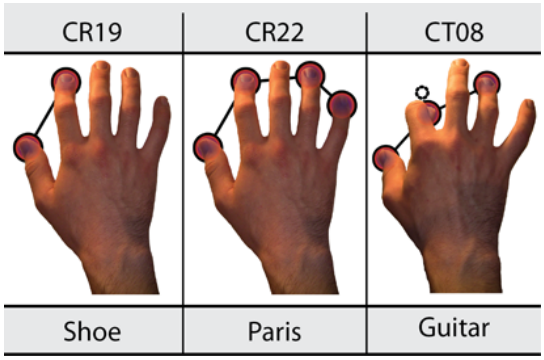


Figure 65: Chord to command mappings for which users have reported spontaneous mnemonic strategies.

5.5.8 *Qualitative Results*

In the post-experiment questionnaire, we asked participants to tell us about the mnemonic strategies they used to memorize chording gestures and their mapping to commands. Several participants reported using semantic associations between the posture of the hand when performing the gesture and the command name (see Fig. 65). For example, the gesture for "guitar" was "like striking a guitar string" (reported by one participant in the CheatSheet group, and two in the Arpège group), the "Paris" gesture could represent "the Eiffel tower" (reported by two participants in the Arpège group), and the "shoe" gesture was seen as "the gesture for tying shoes up" (reported by one participant in the CheatSheet group, and one in the Arpège group). In total, five participants in the CheatSheet group and eight participants in the Arpège group reported using such semantic associations. In the two groups, we respectively counted 11 and 20 semantic associations used by these participants.

Other participants reported explicitly memorizing the geometric shape of the set of points on the surface (four participants in the CheatSheet group, and one in the Arpège group). Some participants were unable to tell how they memorized the chords.

Two participants from the first group complained about the cheat sheet as conveying too much information at the same time. In the second group, four participants spontaneously told us that they were impressed by their progressive skill acquisition with *Arpège* (regarding memorization and ease-of-use). We did not get such observations from the participants in the *cheat sheet* group.

5.5.9 *Discussion*

In this experiment, we observed that in both groups, participants rapidly learned an *expressive* vocabulary of chording gestures and stopped using help, which validates our first hypothesis —i.e. "par-

ticipants will be able to learn and memorize a representative set of chording gestures". During the third session of the first day, they used help less than 5% of the time, and gave appropriate answers to the confirmation questions more than 90% of the time. Furthermore, regarding mid-term memorization, the *recall rates* in both subsessions of the second day are not significantly different from the high performance they have reached at the end of the first day.

The comparison between the two techniques show that participants did not learn and memorize chords significantly better with *Arpège* than with the cheat sheet. Therefore, hypotheses (ii) —i.e. "learning will be faster with *Arpège* than with the cheat sheet"— and (iii) —i.e. "mid-term memorization will be better with *Arpège* than with the cheat sheet"— are not supported by these results. However, the cheat sheet we designed and the associated triggering system have several advantages compared to the cheat sheets described in the literature: it shows a real picture of a hand, giving clues as to the appropriate posture of the hand for performing the chord; it provides additional information about the relative finger positions (dashed white circles representing the relaxed positions as a reference for the tense positions); and it is displayed without requiring the user to navigate in a menu or switch context. Thus, we can only state that learning with a dynamic guide was not significantly better than learning with an adapted static learning method in this experiment.

Hypothesis (iv) —i.e. "participants learning chords with *Arpège* will have more chords recognized by the classifier than participants learning chords with the the cheat sheet"— is not supported either by these results. Nevertheless, as in the first experiment, the experimental design did not allow us to identify errors from the device, errors from the participant and errors from the recognizer. A specific study must be conducted to improve the results of our recognizer and find a setup to distinguish between these types of errors.

Although participants from the two groups reported spontaneous mnemonic strategies, most of them were in the Arpège group (eight participants vs. five in the CheatSheet group). It suggests that using *Arpège* might help users make sense of chording gestures and internalize them. This result is encouraging since participants had an overview of the whole vocabulary with the cheat sheet, although with *Arpège* they focused on one chord at a time.

5.6 SUMMARY AND PERSPECTIVES

This chapter addressed the problems of designing and learning *expressive* vocabularies of chording gestures. We distinguish between "relaxed" chords —where fingers are not flexed, adducted or abducted— and "tense" chords. We introduced two guidelines and

presented a design space to help designers create *expressive* vocabularies of chording gestures that respect the biomechanical constraints of the hand, thus remaining *operable*. In a first experiment, we explored our design space and collected subjective evaluations from casual users on *understandability* and comfort of use of chording gestures. "Tense" chords are less *understandable* and *operable* than "relaxed" chords, although our cheat sheet displaying additional information about proper finger positions supports a sufficient level of *understandability* for all chords. We validated our first guideline —i.e. "Avoid chording gestures where either the middle or ring fingers are lifted while the neighboring fingers are touching the surface"— for "relaxed" chords, while for "tense" chords, the number of fingers involved in the chord and the number of fingers in tense positions seem to have more importance regarding perceived *comfort*.

Our next challenge was to assist users when performing and learning chords, especially the "tense" chords. We described *Arpège*, a dynamic guide for chording gestures providing a progressive feedforward showing how to perform chords, and a progressive feedback on the appropriateness of finger positions. We ran a second experiment to investigate the learning and mid-term memorization of chording gestures as command triggers. The results suggest that our *expressive* vocabularies of chording gestures can be efficiently learned and memorized by casual users. Chords are recalled as efficiently with *Arpège* as with our cheat sheet providing additional information on the relative positions of fingers. Furthermore, *Arpège* shows precious advantages for the *learnability* of chording gestures:

- it breaks down the complexity of chording gestures for novice users;
- it ensures that the user performs the chord appropriately and avoids uncomfortable hand postures;
- it supports error correction;
- it allows casual users to explore the vocabulary of commands;
- it takes advantage of partial memorization by allowing intermediate users to focus on the parts of the chords for which they still need assistance.

Such advantages might help the transition from novice to expert when using *expressive* vocabularies of chording gestures. *Arpège* also increases the *understandability* of chording gestures, since it minimizes visual complexity and provides additional visual features to make sense of chords, such as colors. Such improvement in *learnability* and *understandability* might explain why more participants learning chords with *Arpège* in the second experiment reported that they have spontaneously built strong mnemonic strategies.

From the point of view of interaction design, *Arpège* can be used on tablets, since it does not require as much screen space as a cheat sheet. In addition, our on-line learning method prevents users from switching context to learn chords from deported graphical representations, menus, configuration tools or manuals.

We outline some areas for future work on chording gestures and *Arpège*. In the second experiment, we tested *Arpège* with a set of 12 chording gestures. We plan to study how the technique scales, i.e. the effect of increasing the *semantic width* of the gesture set on *Arpège*'s efficiency. Furthermore, since *Arpège* can favor access to expertise, we want to run long-term studies with *Arpège* and evaluate its efficiency for particularly complex chording gestures. Finally, although our chord classifier does not require training and provided satisfying results, we did not focus on recognition rates. The efficiency of the classifier can be improved by taking into account the geometric rules we use for designing chording gestures. Its performance should also be tested more formally.

6

THE "MATERIALITY" OF MUSIC SOFTWARE: A DESIGN FRAMEWORK FOR UNDERSTANDABLE AND EXPRESSIVE MAPPING STRATEGIES

> *"Can we define the term musical instrument? It is impossible, as well as we cannot state any precise definition of music that would be valid in every situation, every period, and every use of this art. The problem of instruments rejoins the question of the boundaries of music. An object can be sonorous; how and why can we say it is musical? For which kind of qualities Music will promote it to the same grade as others instruments?" (André Schaeffner, 1936, "L'origine des instruments de musique: Introduction ethnologique à l'histoire de la musique instrumentale", Payot)*

In this last chapter, we study music software from the point of view of our framework on the *usability* and *expressiveness* of interactive systems.

We argue that grounding the design of music software in some essential parts of musical instruments can (i) make music software more *accessible* to novice users, (ii) provide the *expressiveness* which is required to develop expert skills and (iii) lead the audience to a deeper *understanding* in the context of live performances.

Claude Cadoz observes that musical instruments have been created to allow humans to produce sound with the hand, which is the part of the human body having the most complex interaction with the environment (Cadoz, 1999). Musical instruments act as intermediaries between the slow movements of our hands, and the rapid oscillations of air pressure that can be sensed by our eardrums. The resulting sounds complement these of the human voice and are based on various physical phenomena (ibid.).

Since the invention of *MUSIC* –the first widely used sound generation software– by Max Matthews in 1957, the computer has progressively become a tool for music creation and performance. However, due to the proliferation of experimental projects aiming to define new ways to create music with computers, this activity lacks standardization and the distinction between instrument-making, compo-

sition and playing –which is a key aspect of the richness of acoustic instruments– is not clearly established. The absence of sustainable practices does not allow non-musicians to build an advanced and stable knowledge about the use of music software, which might restrict their *access* to such systems.

Furthermore, as the advent of computer music mostly comes from the evolution of software sound synthesis, computer musicians who *play the computer* often focus on controlling the variables of synthesis and processing algorithms, making music software mostly *present-at-hand*. The relation between computer musicians and their software is less embodied than with acoustic instruments, which limits non-musicians' *understanding* [1] of performances. In 1992, Marc Battier stated that "traditionally, to attend a music performance is to apprehend through the sight the intention which is loaded in the instrumentalist's gesture. In the mediation of the technological work, this prediction does not work all the time" (Battier, 1992). Today, twenty years later, it is still not rare to see computer musicians hidden behind their laptops on stage, controlling sound production with mice, keyboards, buttons and sometimes potentiometers. Such gestures rarely convey information about temporal, dynamic, harmonic and tonal aspects of the music they produce. In turn, the sound created by computer means rarely reflects the gestures and effort of the performer. Although creating musical sound is an art by itself —as defended by Pierre Schaeffer who introduced *musique concrète* in 1948 and wrote *Le solfège de l'objet sonore* (Schaeffer, 1966)—, *playing* music has historical and social grounds which should not be neglected in the field of computer music either.

Several authors have conducted studies showing the importance of parameters *mapping* in music software (Hunt and Wanderley, 2002). The structure of these links between input devices and functionalities of the music software dramatically influences their *understandability*, *learnability*, *accessibility* and *efficiency*. Although ad-hoc developments have proven very successful and usable, e.g., The Reactable (Geiger et al., 2010), and while some innovative musicians manage to create complex instrumental gestures with non-expressive devices [2], our work is more in line with the definition of a *lutherie* and *organology* of computer music (e.g. Jordà, 2005). Indeed, despite existing studies on input devices inspired by morphological aspects of musical instruments (e.g. Haury, 09; Maruyama et al., 2010) and on software synthesis inspired by instrumental sounds (Smith, 2010), various aspects of the experience of instrument playing are not matched in the field of computer music. However, according to Marc Leman, "it is

1. Here, we talk about understanding as presented in chapter three, including embodied perception of structural and expressive aspects.

2. See for example Jeremy Ellis using Native Instruments' Maschine: www.youtube.com/jeremyellismusic

assumed [...] that the detailed study of the way in which performers handle acoustic instruments may reveal basic components of an embodied interaction and communication pattern that can be exploited for the development of electronic interactive systems" (Leman, 2007).

In line with the framework developed in chapter three of this dissertation, we first review studies on the structural, functional and instrumental properties of acoustic instruments to identify the core aspects of their *usability* and *expressiveness*. We then review research on music software for live performance, and underscore some aspects of instrument playing which have been neglected in computer music. Based on our description of acoustic instruments, we provide insights into the success of dynamic mapping strategies. We then define *mapping through behavior models*[3] (MBM) as a *class* of music software and provide a design framework and a software architecture to describe existing systems and create new designs. A *behavior model* is the dynamic core of the mapping layer, which is controlled by the user and in turn controls audio synthesis. Its autonomous evolution and its reactions to stimuli are defined by various rules. Its visual representation is designed to help novice or expert musicians, as well as the audience, make sense of its functioning, and thus compensate for the lack of *materiality* of music software. By focusing on the embodied interaction of computer musicians with music software and by providing graphical abstractions of the underlying *behavior* of the interface, our goal is to improve the *understandability* of computer music systems and therefore increase their *accessibility*. We then present two implementations within our a software architecture and provide guidelines for further designs.

6.1 A POINT OF VIEW ON THE ESSENCE OF ACOUSTIC INSTRUMENTS

6.1.1 *Instrumental Properties Of Acoustic Instruments*

Acoustic instruments are at the same time *expressive* –by taking the richness of instrumental gestures into account–, *effective* –by creating numerous sounds–, and *efficient* for musical expression. Considering the theories presented in chapter 2 and 3, musical instruments have a multifaceted nature. First, expert musicians have learned how to use them as functional organs that are mostly "ready-to-hand". They know precisely which sounds can be produced, as well as the gestures

3. This work is the result of a collaborative effort with Vincent Goudard, Hugues Genevois and Boris Doval from the *Lutheries, Acoustique, Musique* (LAM) team of Université Paris 6. The contributions in sections 2.3.2 to 2.3.6 have been published at the Sound and Music Computing conference (Goudard et al., 2011a) and at a workshop on music software, JIM 11 (Goudard et al., 2011b)

they must perform to achieve these sounds (Cadoz and Wanderley, 2000).

In order to support the development of such an expertise, musical instruments feature a high level of consistency from various points of view (Cance and Genevois, 2009). Temporally, gestures and resulting sounds are simultaneous (Cadoz, 1999). Spatially, the sounds are directly produced by the artifact which is operated by the musician, and the sound radiating from each instrument has a unique directivity (Genevois and Ghomi, 2010). Causally, the dynamic and spatial aspects of the musician's gestures are legible in the resulting sounds. Such an *accountability* helps musicians structure their actions and perception of the tactilo-proprio-kinesthetic (e.g., vibration, displacement of movable parts) and auditory feedback (Kululuka, 2001). We observe that the *learnability* of acoustic instruments is not only ensured by adapted learning methods, but also by the *predictability* (Pelinski, 2005), *reproducibility* and *variability* of their musical output (Jordà, 2004): musicians learn to play music by reproducing and progressively modifying more and more complex musical sequences (O'Modhrain and Essl, 2004a).

As a result of the rich tacit knowledge gained by musicians when playing, their interaction with instruments is deeply embodied and complex. In the literature, musical instruments are often described in terms of *intimacy* –"the perceived match between the behaviour of a device and the operation of that device" (Fels, 2004)– or *transparency*. For Fels et al., "transparency provides an indication of the psychophysiological distance, in the minds of the player and the audience, between the input and output [...]. Full transparency for the player means that the device's output exactly matches the player's expectation and control" (Fels et al., 2002).

Second, the structural and functional properties of acoustic instruments have been refined over centuries, according to the aesthetics of music (Baily, 2001; Jordà, 2005). This evolution is grounded in the co-adaptation between their morphology, music practices and music aesthetics. Levitin et al. insist that only the instruments which showed sufficient expressiveness, clarity of control alternatives and pleasing timbres have survived (Levitin et al., 2002). Nevertheless, as underscored by Dahlstedt, there is a tradeoff between polyphony and richness of control in acoustic instruments (Dahlstedt, 2009). For example, numerous notes can be played simultaneously on the piano, while the indirect interaction between the musician and the strings restricts control over timbre. In contrast, the violin allows fine continuous transitions between pitches and expressive articulation, but playing chords is complicated. Several authors also identify a tradeoff between *challenge*, *frustration* and *boredom* (Csikszentmihalyi, 1991). For example, the kazoo is easy to play, but its possibilities are quickly exhausted.

The sound of acoustic instruments reflects the physical effort of the musician (Winkler, 1995). For example, the timbre of instrumental sounds often depends on loudness and thus on the energy provided by performers. Furthermore, the physical construction of musical instruments reflects the appropriate instrumental gestures, and participate in the nature and quality of sound (Bertelsen et al., 2007). Bertelsen et al. describe these complementary aspects as the *materiality* of acoustic instruments.

Thanks to these functional and structural properties, even non-musicians having only an implicit knowledge of musical structure can understand the relationship between the musician's gestures, the morphology of instruments and the resulting sounds (Fels et al., 2002). We call that aspect the *readability* of instrumental gestures, which depends on the *consistency* and *accountability* of instruments, as well as on the cultural knowledge about traditional instruments gained by the audience. Materiality and readability of instrumental gestures are also crucial for coordination in collaborative music production in bands and orchestras (Arfib and Kessous, 2005).

Third, acoustic instruments allow both novice and expert activities, since non-musicians can play a few chords along with a song with little practice. Wessel et al. illustrate this duality by describing musical instruments as having a "low entry fee with no ceiling on virtuosity" (Wessel and Wright, 2001). However, some instruments require a non-negligible expertise to play a single note properly (e.g., various wind instruments, fretless string instruments, trumpet), while others provide a direct access to notes and allow absolute non-musicians to improvise simple albeit beautiful music (e.g., keyboard instruments), as pointed out by Jordà (Jordà, 2004). Within our framework on *usability*, virtuosity can be characterized by the absence of breakdowns related to *understandability* and *operability* in the expert use of *expressive* instruments, while the "low entry fee" corresponds to what we call *accessibility*.

The complex nature of musical instruments and the importance of the cultural and social aspects of music make it difficult to grant the status of *musical instruments* to computer systems. In 1988, Cadoz introduced the term "instrumental gestures" to describe the gestures of the musician (Cadoz, 1998). Cance et al. reported linguistic studies confirming that a "shift from instrument as an entity to instrumental quality" occurred during the twentieth century, and proposes to study the "instrumentality" of computer systems (Cance et al., 2009). For music software, this quality can be evaluated in the context of use and depends on the emergence of embodied interaction, standard practices, and on the development of *repertoires*. Cadoz distinguishes between three categories of instrumental gestures used to play acoustic instruments (Cadoz, 1998): excitation gestures, of which physical

energy is used by the instrument to produce acoustic energy; modification gestures, by which the musician affects the structure and properties of the instrument (e.g. using pedals on a piano, moving the left hand on the fretboard of a guitar); and selection gestures by which musicians select components of the instrument (e.g. playing one of the strings on a violin).

6.1.2 *Functional Decomposition Of Musical Instruments*

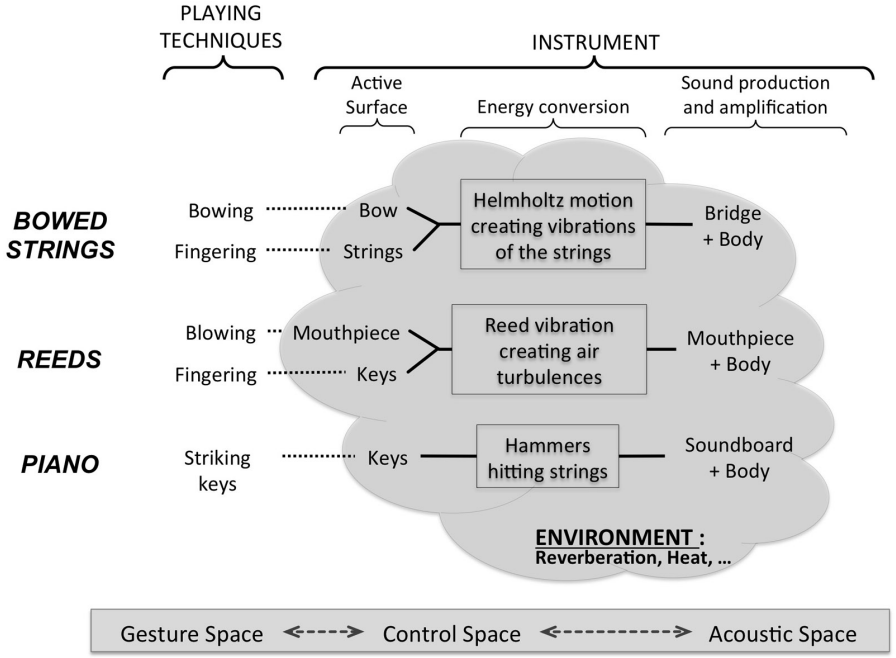


Figure 66: Functional decomposition of the gesture-to-sound transformation operated by acoustic instruments. The "active surface", energy conversion, sound production and amplification mechanisms are designed and built by the instrument-maker. Playing techniques are used by musicians, and the environment brings additional perturbations to the behavior of the musicians and to the sound.

In this section, we present a functional decomposition of acoustic instruments. The goal is to identify and reuse the essence of their functioning in order to support the high level of *expressiveness* and *understandability* presented in the previous section.

For Claude Cadoz, musical instruments are artifacts which transform instrumental gestures into sound in real-time. In the interaction between the musician and the instrument, "specific phenomena are produced, which behavior and dynamic evolution can be mastered by the subject" (Cadoz, 1999). Hugues Dufourt presents a similar idea, asserting that "the musical instrument is more than a mere machine. It is an energy conversion device with expressive goals" (Dufourt, 1995). This energy conversion is operated by the physical mecha-

nisms which are at play in musical instruments. For example, the mechanism of hammers in the piano allows the conversion of the energy transmitted by pressing keys into percussive motion of hammers on strings. In bowed strings, the Helmholtz motion allows the conversion of the translation of the bow into the vibration of the string. This energy conversion is the core of musical instruments and the center of our functional decomposition (see Fig. 66), allowing musicians to produce sounds with physical phenomena that are not directly possible to produce with the human body (e.g., sustained vibration of strings, managed air turbulences in the body of wind instruments, vibrations of flat surfaces in vibrating membrane instruments). Note that these physical phenomena are non-linear, thus participating to the richness of instrumental sounds. In fact, all acoustic instruments introduce non-linearities, in the vibration phenomena or in relationship between musical parameters (e.g., loudness and timbre in reed instruments) (Cadoz, 1999).

The next block of our decomposition addresses sound production and amplification (see Fig. 66). In the case of the piano, the soundboard is linked with the strings and amplifies their vibration. Similarly, the body of the violin is a "sound box" coupling the vibration of strings to the surrounding air, making it audible.

When playing music, the gestures of the musician are received by the "*active surface*" of the instrument (Baily, 2001), which John Baily defines as the "contact points with the body". This part of our decomposition is where musicians input energy into the functional properties of instruments, with various playing techniques (see Fig. 66). For example, the *active surface* of the piano is made of the keyboard and pedals. As seen in the previous section, that *active surface* conveys information about the gestures which are taken into account by the instrument. In most Western instruments, the *active surface* is the result of centuries of experimentation and combinations in order to find the best input method to control the energy conversion and sound production mechanisms of the instrument, with its inherent possibilities and restrictions. For instance, the first organ was created in 270 BCE, then the keyboard was created to control the simultaneous production of several notes. 1600 years of innovation in instrument-making were necessary to create the keyboard which is used on today's pianos. The advantage of the piano, with its internal mechanisms and active surface, is to allow a wide range of dynamics and a sufficient decoupling between the musicians and strings to play several notes at the same time. Furthermore, this *active surface* has also been adapted to the internal mechanisms of various other instruments: when coupled with the energy conversion mechanism of the harmonica it led to the harmonium, and combining it with plucked psaltery led to the harpsichord.

Given the *materiality* of acoustic instruments, creating music is not only structured according to auditory schemes, but also according to movement schemes (Baily, 2001). Today's instruments represent the most efficient combinations of *active surfaces*, energy conversion mechanisms, sound production and amplification. The interaction between these two latter elements delineate the reachable acoustic space, which corresponds to the timbre of the instrument and the notes and chords it can produce. The *active surface* of the instrument and adapted playing techniques define the gesture space, by favoring some gestures (Baily, 2001).

Therefore, the resulting morphology of the instrument, including the *active surface*, acts as an abstraction of its internal mechanisms. The musician can create music by playing the instrument without being aware of its internal behavior. When playing the piano, the musician just needs to manipulate keyboards and pedals, and perceive the various resulting feedback. The consistency of the link between the different stages of our decomposition provides enough information to ensure *understandability*.

Finally, we also consider the influence of the environment on the perception and action of the musician. Many instruments are indeed made up of materials that react to changes in humidity or temperature, and performances is influenced by the reverberation of the room in which the instrument is played.

We argue that applying such a decomposition —centred on internal behaviors shown to the musician as abstractions— and such processes of instrument-making —including experimenting combinations of *active surfaces*, energy conversion mechanisms and sound production mechanisms— to music software can help creating designs that are at the same time *usable* and *expressive*.

6.2 THE ADVENT OF COMPUTER MUSIC

6.2.1 *The "Reduction Of Feel"*

By allowing transmission and recording of sounds, the invention of the telephone and the phonograph disturbed the relationship between humans and music (Cance and Genevois, 2009). Then, sound synthesis introduced a complete decoupling between the energy inputed by the performer's gestures and that of the acoustic vibrations which are produced. Cadoz explains that these systems behave like relays since they present two distinct energetic chains (Cadoz, 1999). The temporal, spatial and causal consistency of such systems —including energetic and functional aspects— is not ensured. The consequences of this decoupling between musicians and sound production have been widely studied by Cadoz et al. (Cadoz, 1999; Cadoz and Wanderley, 2000). Various other authors have pointed out the

lack of embodiment in the relationship between computer musicians and music software (e.g. Dahlstedt, 2009). Therefore, listeners' *understanding* of the actions of the musician is limited (Paine, 2009). Curtis Roads summarized this situation by calling it "the reduction of feel" in music software (Roads, 1996).

But this new gap between the physical energy of the musician and the acoustic energy produced by computer means also has considerable advantages. It widens the reachable acoustic space by creating sounds that do not exist in nature. In fact, we identify a new tradeoff between the *effectiveness* of music software —with synthesis engines [4] aiming to cover a wide range of sounds— and its *expressiveness*, compared to musical expression with acoustic instruments.

The study of parameters *mapping* in computer systems addresses this point and is described as a central issue by several researchers (e.g. Rovan et al., 1997; Hunt et al., 2002; Van Nort and Wanderley, 2006).

6.2.2 *Mapping Strategies*

Several concepts have been created to describe mapping strategies between the parameters provided by input devices and the parameters of the synthesis engine to control. Above all, the role of mapping is to reduce the control of synthesis engines to parameters which users can understand and operate. For example, if we want to control the perceived distance of a sound with just one potentiometer, it must be mapped at the same time to the loudness, delay and reverberation of that sound. Indeed, a distant sound is perceived as less loud, takes more time to reach the ear, and the reflected sound becomes more important compared to direct sound. So, in many cases, one wants to control an aspect of the sound that corresponds to several synthesis parameters or, on the contrary, to combine input parameters to control a single synthesis parameter. Rovan et al. (1997) categorize these mapping strategies as "one-to-one", "convergent" and "divergent" (see Fig. 67). Hunt et al. (2002) call these latter two categories "many-to-one" and "one-to-many".

Kvifte et al. (2006) talks about "parameters coupling" and underscores the fact that the input parameters or the synthesis parameters can be coupled. For example, various dimensions of an input device can be combined, and synthesis parameters can influence one another (as it is the case with loudness and timbre in reed instruments). Goudeseune (2002) presents a similar approach, by considering "primary" input parameters –directly linked to synthesis parameters–, and "secondary" parameters, modifying or even inhibiting the role

4. In analogy to *graphics engines* used for computer games development, we call *synthesis engines* the algorithms of digital synthesis and processing in music software.

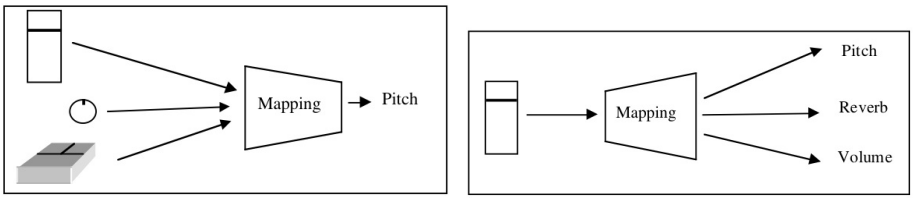


Figure 67: Convergent and divergent mappings. Pictures from Hunt and Kirk (2000)

of the former ones. For example, the gain faders on a mixing console give access to primary parameters, while the mute buttons of each track control secondary parameters.

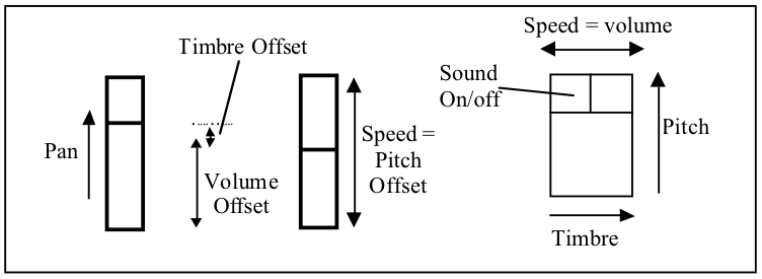


Figure 68: The multiparametric mapping tested by Hunt et al. (2000): two potentiometers and a mouse are used to control the pitch, timbre, volume and triggering of a sound

Hunt et al. distinguish between "simple" —i.e. one-to-one— and "complex" —i.e. many-to-many— mapping strategies (Hunt et al., 2000). In the experiments reported by the authors, "simple" mapping are more *accessible*, while "complex" mappings are more *expressive* as well as *effective* for complex tasks. Participants were more "satisfied" and "engaged" with the *multiparametric* complex mapping (see Fig. 68). They learned better over time, being able to internalize the control of the music software. The interviews revealed that the embodied perception of sound was richer with "complex" mappings, since participants were able to think "gesturally" (Hunt et al., 2000). The authors state that the feel of *expressiveness* was related to the challenge brought by "complex" mappings.

Since the musical result can depend both on the input parameters and on the computational algorithms of the system, various authors propose to differentiate between "static" mappings and "dynamic" mappings, which evolve over time and/or analyze input parameters to provide higher-level description of musicians' gestures (Arfib et al., 2002; Momeni and Henry, 2006; Dahlstedt, 2009).

With all these definitions ranging from simple connection to dynamic behaviors, the exact contours and boundaries of the concept of mapping remains unclear. Furthermore, to our knowledge, no map-

ping strategy has been designed to take advantage of the richness of the experience of playing music with acoustic instruments.

6.2.3 *The Materiality Of Computer Music Systems*

In 2007, Bertelsen et al. studied the importance of the "*materiality*" of interaction instruments, especially for creative activities (Bertelsen et al., 2007). They conducted interviews of two Max/MSP[5] expert users. The two composers expressed that, due to certain particular properties, the software was "an integrated part of their creative process" (ibid.). The feeling of *materiality*, according to the authors, is related to two complementary aspects of the strong and *understandable* structure of Max/MSP: (i) the limits and resistance imposed by the software; (ii) the guidance provided by the software by favoring certain practices. The combination of these two aspects is seen by these composers as fundamental for the creative process. The authors compare this situation to playing music with acoustic musical instruments, where the physical and functional structures of the artifact drive its use and create a challenge that is necessary for highly creative tasks. Then, the constraints inherent to such materiality are understood by musicians as a positive and even necessary basis for creativity.

Thanks to their engagement, musicians accept that "they have to discipline themselves in the use of the software in order to benefit from its complexity, but most important also to extend or perhaps even transcend its limitations. Like a violin, such software is not easily mastered, but with enough work it provides a possibility for achieving virtuosity." (Bertelsen et al., 2007). The fact that Max/MSP is alternatively *present-at-hand* and *ready-to-hand*, as acoustic instruments when used in highly creative activities, leads to a deeply embodied interaction with the software.

In an article on the structure of acoustic instruments, John Baily shows that the structural and functional properties of two traditional Afghan stringed instruments –the rubâb and the dutâr–, with their inherent possibilities and limitations, had an important influence on the creation of the associated repertoire (Baily, 2001). Therefore, these constraints are not drawbacks of the instruments, but rather necessary aspects for the creation and structuring of new music genres. He concludes that all the rules of folk blues guitar carry within themselves the structure of folk guitars and the footprint of the work of musicians and composers who tended to make the best use of the richness of that instrument (Baily, 2001). This is how music genres and playing techniques have co-adapted to the morphology of instruments.

5. Max/MSP is a leading visual programming environment for music and graphical creation, developed and maintained by Cycling 74: cycling74.com/whatismax/

The concept of *materiality* seems to explain the positive feedbacks provided by users in the study on complex mappings conducted by Hunt et al. Hunt et al. (2000). We identify two complementary design strategies aiming to improve the materiality of interaction instruments: metaphors, and the concept of *metonymy* introduced in HCI by Bertelsen et al. (2007).

6.2.4 *Metaphors*

An exhaustive review of the use of metaphors in interaction design is provided by Blackwell (2006). For Blackwell, a User Interface can be considered as an abstraction of the functional properties of computers. Metaphors help users *understand* and *learn* how to interact with unfamiliar objects, thus increasing the *accessibility* of computer systems. In order to take advantage of the knowledge already gained by common users, interaction designers introduced a set of metaphors, including "menus" –as menus in restaurants–, "carriage return" –referring to typewriters–, "buttons" or "scrolls" in document, analogous to parchment or papyrus where the bottom of each page is glued to the top of the next (ibid.).

Metaphors were first described as the ground for embodied sensemaking by Lakoff and Johnson (1980), which is illustrated by all the spatial and physical metaphors we commonly use in language, referring to bodily and physical experiences. For example, we use image schemata related to physical and bodily situations –e.g. center-periphery, part-whole, containers– or with orientation and movement –e.g. near-far. In fact, tacit knowledge is first based on action and not on abstract mental processing, as presented in chapter two and validated in studies on child learning (Antle et al., 2009).

In HCI, we can nevertheless complain about the exclusive use of metaphors for the design of the visual aspects of user interfaces. Antle et al. state that non-visual metaphors should also be used (ibid.). Metaphors of actions and interaction might dramatically improve the *understandability* and *learnability* of interactive systems, since interaction with computers is not exclusively visual, but also enabled by body movements, gestures and physical object manipulations: "Metaphorical interaction models may be able to support the user to intuitively enact appropriate input actions and understand the relationship to resulting system responses" (ibid.). Antle et al. insist that designing such metaphors requires that interaction designers are aware of how tacit knowledge is gained by users. Although the authors do not provide further description of such mechanisms, the embodied and social sources of tacit knowledge have been referred to in chapters two and three of this dissertation.

In music, teaching often use metaphors to refer to instrumental gestures. Goormaghtigh presented a study on the finger tech-

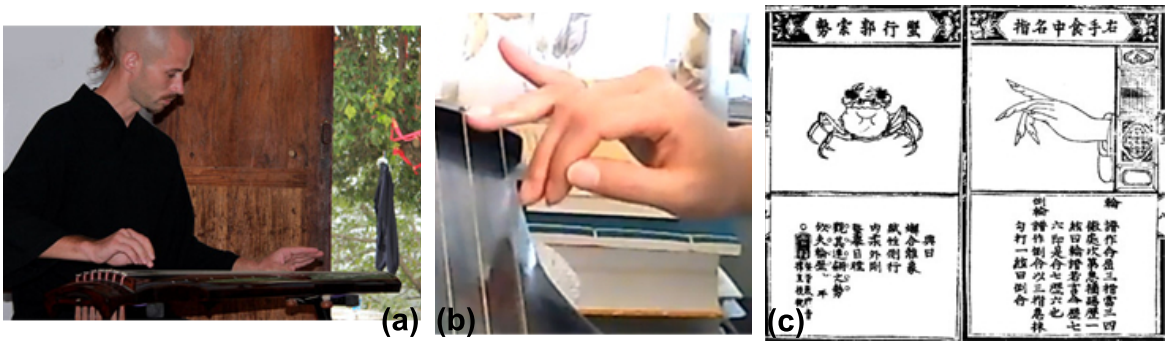


Figure 69: Body posture and finger techniques for playing the Guqin (a) are taught with metaphors. The "Lun" technique (b) is compared to the behavior of a crab (c) (pictures taken from: Jakob Isaksson's blog (a); Peiyou Chang Youtube page (b); the *Taiyin Daquanji* book on Guqin technique)

niques of the Chinese Guqin [6], which is an extreme example of using metaphors of movement to teach instrument playing (Goormaghtigh, 2001). This 5000 years old stringed instrument is played with both hands while laid across the player's lap (see Fig. 69(a)). The traditional playing technique includes more than a thousand complex finger techniques involving the whole hand to interact with the strings. The teaching method is based on 33 metaphorical images representing the gestures. For example, the "Lun" [7] (i.e. "revolving") finger technique, requires striking a string successively with all the fingernails of the right hand (see Fig. 69(b). In the *Taiyin Daquanji* [8], which is the traditional book on Guqin technique, the player is asked to hit the string "in the style of a crab walking" (see Fig. 69(c)). For each finger technique, the method proposes a poem, to make the player *feel* how to perform the gestures properly.

Metaphors are also used to teach the structure of Guqin music: glissandi are evoked by describing the texture of tree trunks, vigorous articulation is evoked by horse rides, etc. All these aspects are also compared to Chinese calligraphy as an art of movement that traditional musicians knew well. As a result, the Guqin has been recognized as the musical instrument with the most precise finger techniques and the most expressive sound (Goormaghtigh, 2001). The teaching method allows the player to develop a sizable tacit and embodied knowledge about the playing technique, and allows to reach a level of *expressiveness* which can not be achieved by the explicit learning of descriptive rules. Becoming an expert player obviously takes

6. See the performances of Yuan Jung-Ping for an illustration of Guqin playing techniques: www.youtube.com/watch?v=RjPmTTCQvbM

7. See Peiyou Chang performing the "Lun" finger technique at www.youtube.com/watch?v=8abtVEgthRk

8. See the SilkQin website describing and explaining the learning method from the *Taiyin Daquanji*, and providing an inventory of the metaphorical images used to learn the finger techniques: www.silkqin.com/02qnpu/05tydq/ty3.htm

decades, but the musical result conveys much information about the movements and effort of the player, thus enriching the perception and intersubjective *understanding* of listeners.

Such embodied metaphors can be used in interaction design to make users master highly *expressive* interactions, by favoring tacit knowledge compared to explicit learning methods such as the Arpege technique presented in the previous chapter. But even for novice users, several authors complains about the limitations imposed by the common use of metaphors. Although they can support what we call *accessibility*, visual metaphors often lock users in (Bertelsen et al., 2007), limit *expressiveness*, and avoid embodied sense-making, engagement, and thus development of expertise. For Blackwell, visual metaphors mimicking nondigital artifacts do not do justice to users' abilities in sense-making and creative interpretation (Blackwell, 2006). He describes the desktop metaphor as a "dead" metaphor. While constantly improved, WIMP interfaces —even metaphoric— are obstacles for HCI research, as already stated by Beaudouin-Lafon (2004). These interfaces have become "intuitive" since they have now been used for decades, but restrict the imagination of both the designers and users. Pirhonen et al. illustrates this consideration by insisting that metaphors should "stimulate" instead of "simulate" (Pirhonen, 2005). These observations must be taken into account to improve the *understandability* of interaction instruments, by leaving room for interpretation and access to tacit knowledge.

In addition to what we have identified from music teaching to improve metaphorical design, Bertelsen et al. describe a complementary way to increase the "materiality" of interactive systems (Bertelsen et al., 2007). The concept of *metonymy*, borrowed from linguistics, describes users' ability to develop personal understanding of the system by going beyond the limitations of metaphors.

6.2.5 *Metonymy And Creativity*

In linguistics, metonymies are described as the substitution of a word by another one having a material or causal relation of contiguity, rather than substituting it by an equivalent in terms of meaning as metaphors do (Jakobson and Halle, 1956). The typical example is substituting "the crown" for "the king" in a sentence. The status of the crown and the king are not equivalent, but the former is a material attribute of the latter.

In the studies of Max/MSP composers and performers reported by Bertelsen et al., the authors observe that in creative activities, subjects often appropriate the software, and do not necessarily restrict their practice to what the designers have imagined as the common use and reified in metaphors (Bertelsen et al., 2007).

For the authors, users use metonymy in their interaction with software when they focus on aspects that have material or causal relationship with the main activity, as well as when they adapt software structure to their goals. For example, Max/MSP users focus on the quality of sound and timbre even if it is not made directly accessible by the software. They also combine algorithmic features and use generative and automated procedures of composition (Bertelsen et al., 2007). The authors consider such exploration of the materiality of software as fundamental in creative processes.

Bertelsen et al. introduce the use of *metonymy* as a design strategy, whose goal is to "use contiguity and material (metonymic) substitutions instead of metaphoric analogies, or leaving the software open for metonymic displacements of the basic metaphors, thus creating less totalitarian metaphors" (ibid.). Metonymy is thus proposed "as a vehicle for users' appropriation of software" (ibid.).

Although the authors focus on skilled composers, and insist that materiality is crucial for making creative users engage in a more instrumental interaction with music software (Bertelsen et al., 2007), we argue that the sense-making processes of novice users happening when they appropriate software —such as building strong mnemonics as presented in the two previous chapters— can also be considered as creative activities. Therefore, *usability* and in particular *understandability* might be favored by *materiality*, by helping users structure knowledge and interaction. This might also ease the transition from novice to expert by leaving room for appropriation.

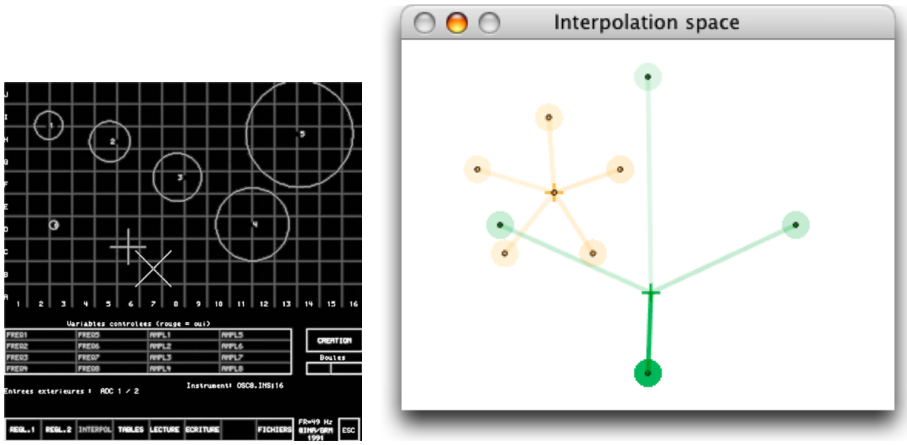
In the following section, we present existing mapping strategies using abstractions and metaphors. We then present our design framework and software architecture, which supports the design of *expressive* music software accounting for the *materiality* of musical instruments by facilitating the creation of *metaphors* and favoring *metonymy*.

6.3 MAPPING THROUGH BEHAVIOR MODELS

6.3.1 *Existing Abstractions And Dynamic Mappings*

In the beginning of computer music, composers had to program the evolution of every synthesis parameter before they could listen to the musical result of their composition. Then, with the evolution of computer systems, synthesis algorithms became *real-time* and input methods started providing a more direct access to sound. However, synthesis algorithms describe sounds in the language of computers, while input devices tend to provide a control which is adapted to human actions (see Roads, 1996). Since the 1990s, advanced dynamic mapping strategies based on abstractions have been introduced to keep computer musicians from the direct control of the parameters

of synthesis engines. We identify three main strategies: interpolation, stochastic systems and mass-spring simulations.



(a) (b)

Figure 70: The user interface of two systems based on spatial interpolation: (a) SyTeR (see Teruggi, 2007); (b) int.lib (Larkin, 2007). In the two figures, the pre-defined states of the system are represented by circles, and the user interpolates between them by moving the crosses.

The first type of abstraction we can find in the literature is based on the spatial interpolation of synthesis parameters. In 1975, the *Groupe de Recherches Musicales* (GRM) introduced the first real-time computer music application based on spatial interpolation: SyTeR (see Teruggi, 2007). Since then, many composers and researchers have gone further in that direction (e.g. Spain and Polfreman, 2001; Bencina, 2005). In such systems, various points corresponding to states of the synthesis engine creating specific sounds are laid out in a 2d space. The musician controls interpolation between these states, and thus controls sound production without directly manipulating the computational parameters of the synthesis engine (see Fig. 70). For Goudeseune, "it reduces the dimensionality of the set of synthesis parameters to the dimensionality of the set of perceptual parameters: it rejects all that the performer cannot actually understand and hear, while performing" (Goudeseune, 2002). It allows the exploration of the reachable acoustic space, without knowing anything about the functioning of the synthesis engine. The system brings limitations and constraints imposed by the nature of the states put in the 2d space, but allows users to explore their own path between these pre-existing states. The combination of these aspects improves the *materiality* of these systems. Some developers have even introduced non-linearity in the 2d space, by providing different influences or *weights* to the states (Larkin, 2007).

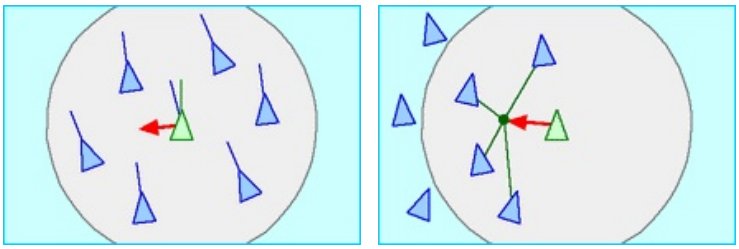


Figure 71: Two steering behaviors of the flocking model: alignment and cohesion (pictures from an article by Craig Reynolds available at http://www.red3d.com/cwr/boids/)

Another abstraction consists in providing statistical models, whose evolution determine the musical result. Such statistical models include the well known stochastic models of composer Iannis Xenakis (Luque, 2009), the Cosmos system created by Bökesoy (2003; 2005), and other widespread particle systems such as Reynolds' "boids", "bird flocks" and "fish schools" (Reynolds, 1987). With the Cosmos model, the user sets the stochastic distribution functions of a self-organizing structure. The flocking model is an artificial life algorithm simulating the flocking behavior of birds. It is based on three simple steering behaviors describing the spatial movement of each particle, based on the positions and velocities of the neighboring particles: separation, alignment and cohesion (see Fig. 71).

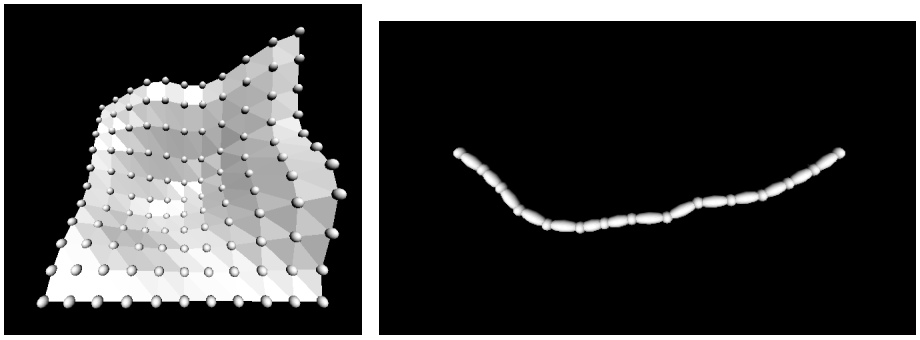


Figure 72: Two structures built with mass-spring models: a "membrane" and a "string" (figures from Henry (2004))

The most widespread kind of dynamic mapping strategies using abstractions is called "physical modeling" (Henry, 2004), based on mass-spring modeling coming from the interaction-mass model formalized and implemented for real-time use in 2003 by Castagne and Cadoz (2003). In such systems, particles evolve in space, every one having a particular inertia and being linked to others according to mass-spring physics equations. Combinations of several particles can create bouncing objects, string movements, chaos or fluid dynamics (see Fig. 72). The behaviors that can be modeled range from real world dynamics to systems with high levels of instability. Then, the

positions and speeds of the particles can be used to control the synthesis of analogous vibration phenomena, or other synthesis engines. The CORDIS-ANIMA model, created by Cadoz in 1978, allows very complex real-time simulations of large mass-spring structures organized hierarchically (Cadoz et al., 2003). Cadoz provides several examples of instrumental gestures simulated with that system, such as the attack of a plectrum or a bow, the vibration of a reed, the behavior of a soundboard or even percussive gestures.

Understandability can be supported by implementing mass-spring models inspired by real world dynamics. Pirro et al. argue that using intermediate physical modeling in mapping strategies supports embodied *understanding* (Pirrò and Eckel, 2011). The observations provided by musicians in the interviews conducted by the authors reveals that they were satisfied by the *materiality* of such mappings. For Pirro et al., this feeling comes from the players' engagement in the challenge brought by the *understandable* limitations of the physical modeling, and from the refinement of control. The authors argue that the engagement and effort of musicians using physical modeling mappings also improves the perception of the audience in the context of live performances. The authors only insist that the physical modeling must be consistent with the produced sounds, which is in line with Castagne and Cadoz criteria for evaluating physical modeling (Castagne and Cadoz, 2003).

However, Modhrain et al. have shown that players easily internalize the dynamic behavior of new musical instruments, even if the mapping is not consistent (O'Modhrain and Essl, 2004b). In such cases, the learning phase is longer but players end up making sense of the dynamic behavior, since they can understand it in an embodied manner and perceive their gestures as schemes of movement and not just as the control of synthesis algorithms.

All these abstractions share the advantage of giving access to a complex musical result controlled by simple parameters. They provide a large amount of output variables and react in real-time. As abstractions, they allow users to explore the acoustic space in an empirical manner. The goal of dynamic mappings is to define consistent "complex" mappings, which have been described as improving *understandability* and access to *expressiveness* —in our terms— by Hunt et al. (2000), as seen in the previous section.

Interpolated mappings can be automated, and the two other kinds of abstractions —stochastic systems and mass-spring models— have an autonomous dynamic behavior, evolving even when the user is not interacting. We call this dynamic evolution exposed to the user's actions the *behavior* of the mapping layer. The main problem we identify is that no common design framework has been provided to describe and evaluate these mapping strategies. Most of the time, each author

uses a particular set of controllers and synthesis engines and implements his own abstractions. We argue that creating a unified methodology for the qualitative comparison of combinations between input, dynamic behavior and sound can match the design of music software to the process of instrument making, and help designers find combinations that maximize *expressiveness* and *usability*.

6.3.2 *A Design Framework For Mapping Through Behavior Models*

The success and richness of the designs presented in the previous section have encouraged us to define a design framework generalizing these ideas, and to propose a completely modular software architecture in order to help designers create *expressive* and *usable* music software, and to establish a more *metonymic* relationship between users and mapping strategies.

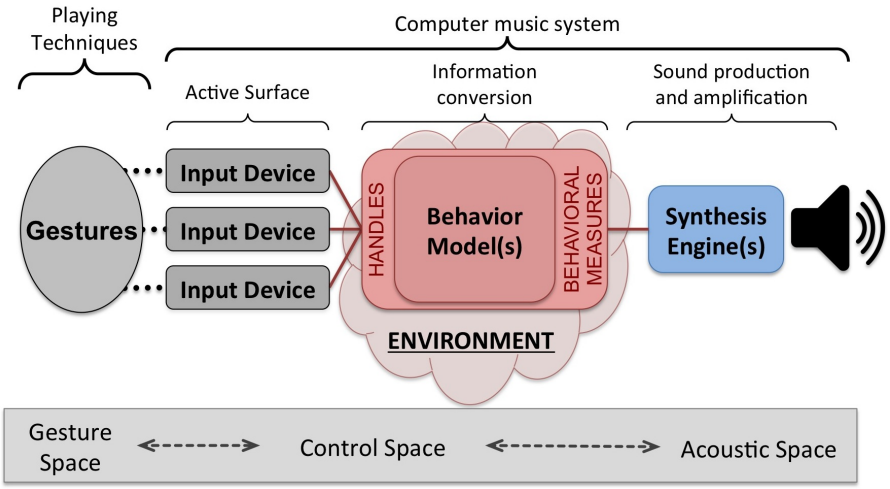


Figure 73: Mapping through Behavior Models for sound synthesis, inspired by the functional decomposition of musical instruments and generalizing the use of dynamic mappings. The active surface is constituted by input devices, and the behavior model converts information. Its output controls the synthesis engine. Here, the environment acts on the part that presents a behavior evolving over time, i.e. the behavior model.

We introduce the concept of *Mapping through Behavior Models* (MBM), involving three types of blocks inspired by the functional properties of acoustic instruments presented in the first section of this chapter (see Fig. 73). First, the input devices play the role of the active surface, receiving the musician's gestures. The *behavior model* represents the internal behavior of the system including information conversion —i.e. reaction to input— and generation —i.e. autonomous evolution. The behavior of this central block is driven by rules defined and implemented by the designer. It represents the energy conversion mechanisms that are at the core of acoustic instru-

ments. It controls the synthesis engine, which generates the audio signal that is sent to the speakers.

In order to clarify the mapping strategies presented in the previous sections, which attribute radically different definitions to the concept of mapping, we propose that the "mapping", strictly speaking, just consists of connections between blocks. The whole dynamic behavior is encapsulated in the *behavior models*. The *behavior model* has "*handles*" –i.e. variables that can be controlled by the *active surface*– and provides "*behavioral measures*" that are used to control the synthesis engine.

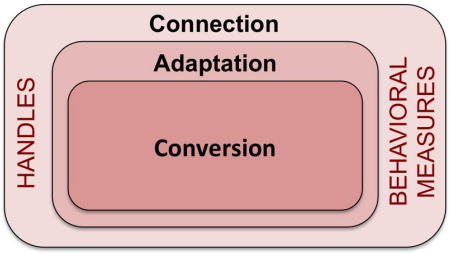


Figure 74: The three layers of a *Behavior Model*

Compared to Momeni et al., who also propose an encapsulation of the dynamic parts of mapping strategies, we provide a more precise definition of dynamic blocks. In fact, *behavior models* are composed of three layers (see Fig. 74). The *connection layer* provides the *handles* and *behavioral measures*. The *adaptation layer* allows the model to translate the incoming information received via *handles* to its own scales and values[9]. The core of the *behavior models* consists of the *conversion layer*, where the model actually evolves and reacts. Its output goes to the *adaptation* layer, where it can be prepared to control the synthesis engine, and is then available for mapping via the *behavioral measures*.

In this chapter, we focus on the building and consistency of the whole system, thus we will not design specific novel input devices or synthesis engines. In the implementations presented in the following sections, we use simple synthesis algorithms since the goal of more complex synthesis engines, such as granular synthesis, is to create complex sounds with little control. Instead, we want to allow an *expressive* control over sound production.

Furthermore, we consider the *environment* as an external dynamic process that can also generate information or disturb the behavior of the *behavior models*. For example, in mass-spring models, an environment can represent external forces, such as gravity, that influence the *behavior* of the system in addition to the links between particles.

9. It is common in music software that the input devices provides values ranging from 0 to 127 —which is the common range of the MIDI norm—, while computation may require other ranges of values.

6.3.3 *Improving Materiality With Visual Representations*

Since the morphology of input devices is not as rich and specific for music production as the *active surface* of acoustic instruments, we insist that visual representations of *behavior models* must be provided. As stated by Baily, visual feedback has an important role in learning how to play acoustic instruments (Baily, 2001). However, visual representations do not have to display the rules of the *behavior models* or the control of the synthesis parameters in an explicit manner, but rather support the *understandability* of the *behavior models* by focusing on information that is related to the perception and understanding of the player, as pointed by Goudeseune (2002). Furthermore, when such systems are played live, visual representations can be displayed to help the audience make sense of the performance.

In light of the studies on music perception and practice presented earlier, we can provide some directions for the design of visual representations. First, they should represent morphological constraints to help musicians feel boundaries and limitations, which are necessary for the sensation of *materiality* presented in the previous section. Second, they should provide a view on the system state and a representation of its reactions to the user's actions on input devices, in order to support accountability. Third, visual representations can be inspired by the *active surface* of acoustic instruments, which are good examples of providing enough morphological information to enable control. For example, pitches can be represented linearly, as it is the case on many instruments such as keyboard instruments. Fourth, some psychological studies observed that children primarily use pictures and abstract symbols to illustrate musical structure (Davidson and Scripp, 1988). Davidson et al. report that children naturally represent structural aspects such as melody, pulse or rhythm with five kinds of symbol systems: "pictorial, abstract patterning, rebus, text, and combinations/elaboration symbol systems". Therefore, we can say that metaphorical representation depicting musical structure or illustrating the effort of the musician —e.g. movement metaphors presented in the previous section— may increase the *understandability* of music software based on *behavior models*.

6.3.4 *Implementation*

We implemented a software architecture in Max/MSP for using *mapping through behavior models*. This implementation is integrated into the Meta-Mallette (De Laubier and Goudard, 2007) and is part of the OrJo [10] (Joystick Orchestra) project. The Meta-Mallette is an environment for real-time control of sound and video synthesis and

10. Webpage of the OrJo Project at: pucemuse.com/orjo/. The implementation of the software architecture and the behavior models presented in the next section has been made by Vincent Goudard during this project

processing. It is compiled with Max/MSP but stands as a *runtime* Max/MSP application and thus has its own user interface and can be used without programming. It includes an SDK and software development conventions. The architecture of the Meta-Mallette is based on *modules* communicating via the Open Sound Control [11] (OSC) protocol. Modules can be loaded at runtime and provide input and/or output parameters. The software comes with a collection of modules which can be selected via the user interface or via OSC messages, and allows their automatic connection —i.e. OSC addresses registration— when loaded.

Figure 75: The user interface of the Metamallette for managing modules. The user selects the modules and mapping presets he wants to load in drop-down lists. He can manage to global parameters of the session on the left (load presets of whole configurations, set the main volume, etc.). And he can further edit the drivers of the input devices and the configuration of the synthesis engines by clicking on the colored buttons on the top.

Our framework led to a new step in the evolution of the Meta-Mallette: the separation between input, mapping, and synthesis. Within the Meta-Mallette, each of the blocks we have defined (see Fig. 73) are independent. New modules created within the Meta-Mallette can be of four types: input modules —e.g. drivers of external controllers such as joystick and pen tablets—, behavior modules —e.g. mass-spring models and interpolation spaces presented earlier—, visual representations and synthesis modules —e.g. additive synthesis, FM synthesis, wavetable synthesis. This modularity even enables to plug a *behavior model* into another one to create higher-level behaviors and test complex combinations of behaviors presented in the litera-

11. Wikipedia entry on OSC: en.wikipedia.org/wiki/Open_Sound_Control

ture. An example of such a combination is presented in section 2.3.6.

Environments are implemented as *behavior models* without *handles*, but with their *behavioral measures* linked to the handles of other *behavior models*.

Visual representations are implemented and connected as Meta-Mallette modules, whose input is controlled by the *behavioral measures* of the *behavior model*, or by parameters of the synthesis engine.

Behavior models and adaptation layers can be saved as Meta-Mallette modules. When loaded, their *handles* and *behavioral measures* become directly accessible from the input and synthesis modules. For example, if a mass-spring model is loaded in the Meta-Mallette, the positions of the input points become accessible from the input modules, and the position and velocities of each point are accessible from the synthesis modules. Combinations between input devices, *behavior models* and synthesis engines can also be saved as presets.

6.3.5 *Advantages Of The Implementation*

In addition to the advantages of using dynamic mappings which were presented earlier, we identify particular advantages of our design framework and software architecture regarding other mapping strategies and studies on instrument playing.

First, *mapping through behavior models* is *descriptive* since it covers the mapping strategies we found in the literature. In our software architecture, "one-to-many" and "many-to-one" mappings or analysis of the input parameters can be implemented by connecting *handles* to *behavioral measures* without defining a dynamical behavior. In this case, our implementation at least eases experimentations by providing an automatic connection between *modules* and by taking advantage of the existing input and synthesis *modules* integrated in the Meta-Mallette.

Our framework can improve the *materiality* of music software: *behavior models* can provide metaphors, and the modularity of our architecture increases the possibility for metonymy in mapping strategies. Users can freely combine input devices, *behavior models* and *synthesis engine* existing in the Meta-Mallette and build unexpected combinations, without programming in Max/MSP. Such modularity is supported by the encapsulation of dynamic behaviors and by the *adaptation* layer improving the polymorphism (Beaudouin-Lafon and Mackay, 2000) of *behavior models*. In fact, the conversion of input and synthesis parameters into the range of values of the *behavior models* is directly embedded into their structure. When a *behavior model* is plugged into other modules and the *adaptation layer* is refined by the designer, the configuration can be saved and reused by musicians.

Therefore, our architecture covers the three types of instrumental gestures defined by Cadoz (Cadoz, 1998). Excitation and modification gestures are used to control the *handles* of a *behavior model*, while selection gestures switch between *behavior models* to control the *synthesis engine*.

The modularity of our architecture also favors the *accessibility* of *behavior models*. In fact, various visual representations can be connected to a *behavior model*. Some can present high-level abstractions related to perceptual or structural parameters, allowing novices to easily *understand* the global result of their actions. Others can illustrate the richness of the behavior more precisely, which is more adapted to an *expressive* use by experts. In the same manner, various input devices can be used to control a *behavior model*, depending on the expertise of the user.

In addition, *environments* can reproduce and enrich the use of *jitters* in computer music, which is a common practice to make sound production less deterministic and thus closer to instrumental sounds.

Finally, the modular architecture we have defined matches the design of music software to instrument-making. In fact, the work of the designer consists of establishing appropriate combinations of input devices, *behavior models*, visual representations and synthesis engines. Such combination can be easily tested in an empirical manner. If a combination allows to cover a wide *control space* (see Fig. 73), we can say that it is *expressive*. If the musician can explore the *acoustic space* of the synthesis engine, the combination is *effective*. Although computer music has not benefited from centuries of co-adaptation like acoustic instruments do, our software architecture makes experimentation much easier and a large amount of input devices and synthesis algorithms already exist which can be easily implemented as Meta-Mallette modules. A designer can easily build and refine the perceived morphology of the system, by creating consistent combinations between *behavior models* and visual representations. He can also test various *playing techniques* on a *behavior model* by changing the input devices with which its *handles* are controlled.

6.3.6 *Examples Of Use And Combinations Of Behavior Models*

In this section, we illustrate the possibilities for combining various *behavior models* within our architecture [12], by describing two models and their combination. Both are geometrical models and are mainly controlled with joysticks. We enriched the visual representation with dynamic aspects, in addition to displaying the geometry of the model.

12. The behavior models we have implemented and the combinations we have tested are presented by Vincent Goudard at vimeo.com/25740547. The mappings described in this section are presented from 1:40 to 4:30 in this video.

We first present the behavior of each model and then describe the mappings we tested for using them to control sound synthesis.

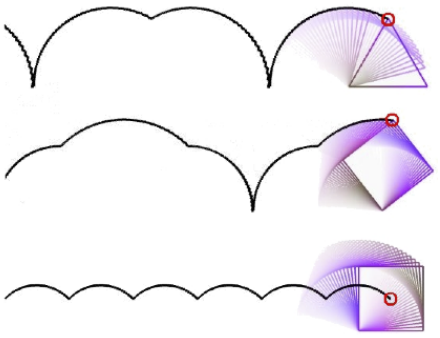


Figure 76: *Roulette* movements and resulting polygon motion

The first one is called "*Roulette*", and is inspired by Pascal's roulette [13]. It is made of a regular polygon that can move in the following ways (see Fig. 76) :

– tipping on one of its corner;

– sliding along one of its sides;

– pendulum motion around an axis.

These movements respond to speed curves in our implementation, which makes this *behavior model* radically different from usual mass-spring physical modeling. In fact, no particular physical constraints are defined in addition to movement, and when a movement is stopped, the polygon will not continue moving since it does not have inertia. We define the "observation point" of *Roulette* as the point whose trajectory we want to follow.

The *handles* of *Roulette* are:

– the number of sides of the polygon;

– the diameter of the polygon;

– the position of the "observation point" on the polygon;

– the type of movement to apply;

– the triggering and interruption of the movement;

– the abscissa and ordinate of the target position;

– the speed curve of the movement.

Its *behavioral measures* are:

– the position, size and orientation of the polygon;

– the position of the "observation point" during the movement;

– the current speed of the "observation point".

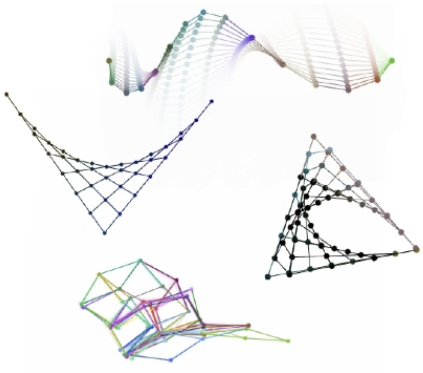


Figure 77: Various Verlet configurations

The second *behavior model* we implemented, called "*Verlet*", is based on the Verlet algorithm [14], which is a particle physics engine. It can simulate a structure made of points connected by elastic links (see Fig. 77). In our implementation, the structure is exposed to an external force, and contained in a box with which it can collide.

The *handles* of *Verlet* and his *environment* are :
- the position, orientation and size of the containing box;
- the length of the elastic links;
- a vector of two values representing the external force in the 2d space.

Its *behavioral measures* are:
- the position of the barycenter of the structure;
- the size of the structure;
- events triggered when collisions with the containing box occur;
- the velocities of the collisions with the box.

In our tests, we controlled the *handles* of *Verlet* with a joystick. Its X and Y axis respectively control the intensity of the external force along the horizontal and vertical axis, which induces movement. Its Z axis controls the length of all the links between the particles. Its *behavioral measures* were mapped to a Karplus-Strong [15] string synthesis algorithm. The size of the structure controls the pitch, while the position of the barycenter alters the timbre of the sound. The collision of any particle with the containing box triggers a noise burst, whose loudness is mapped to the velocity of the collision.

13. This *behavior model* is a generalization of the cycloid curves described by Blaise Pascal in the "*Traité de la Roulette*" in 1659, in his search for a perpetual motion machine: en.wikipedia.org/wiki/Cycloid

14. This numerical integration technique was developed in 1967 by physicist Loup Verlet. Andrew Benson made an implementation for Max/MSP, available at cycling74.com/2010/09/13/jitter-recipes-book-3/.

15. en.wikipedia.org/wiki/Karplus-Strong_string_synthesis

The *handles* of *Roulette* are controlled with a joystick and an additional MIDI controller with faders and buttons. The user defines the target position with the X and Y axis of the joystick, sets the size of the polygon with its Z axis. He selects the type of movement, triggers and interrupts it with the buttons. The additional MIDI device controls the number of sides of the polygon, the position of the "observation point" and the speed curve of the movement. Its *behavioral measures* are mapped to a synthesis algorithm producing a filtered noise [16]. The only parameter used to control the synthesis engine is the movement speed of the "observation point", mapped to the center frequency of the filter.

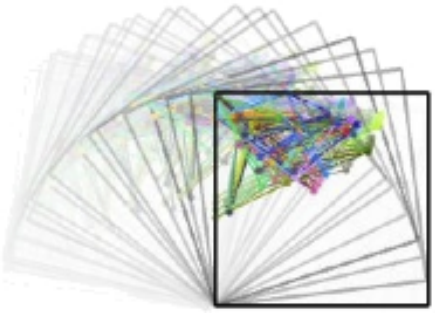


Figure 78: Combination of *behavior models*: the containing box of *Verlet* is controlled by *Roulette*

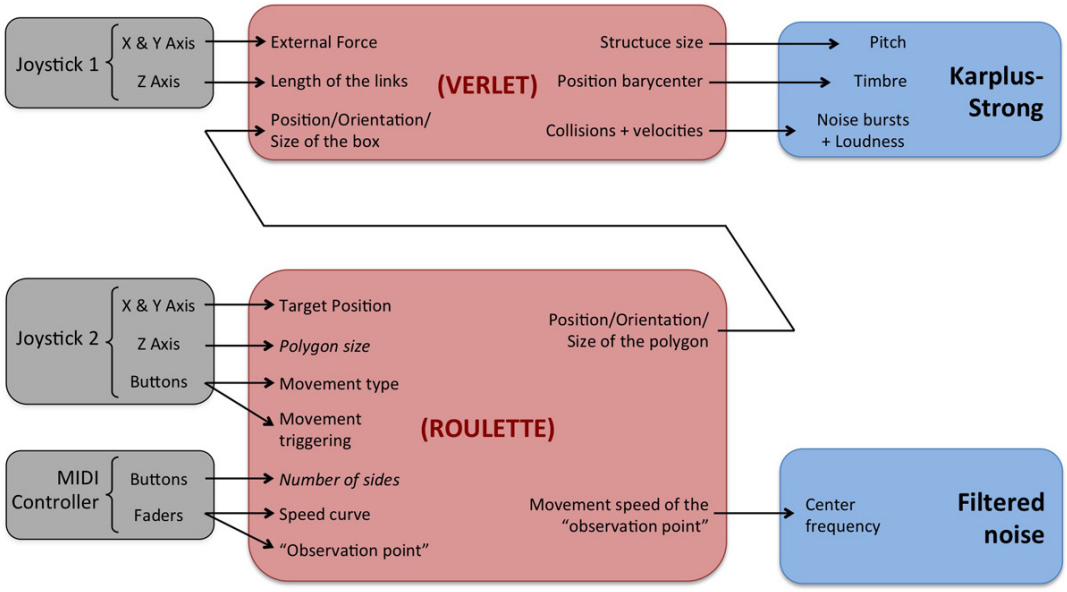


Figure 79: Mapping for the combination of *Verlet* and *Roulette*

We also experimented the combination of these two *behavior models*. In these tests, the containing box of a *Verlet* moves in space according

16. https://ccrma.stanford.edu/ jos/sasp/Filtered_White_Noise.html

to *Roulette*'s motion (see Fig. 78). Two joysticks are used to control the two models. Then, fast movements of *Roulette* give rise to a multitude of collisions between *Verlet* particles and the box in which they are embedded. This mapping is illustrated on Fig. 79.

Although we did not run any evaluation of these models, one of our current goals is to define a taxonomy of existing dynamic behaviors that can be implemented as *behavior models*, and to implement all of them in the Meta-Mallette for further testing and comparisons.

In this section, we have provided an example of an implementation of *behavior models* within our software architecture. The combination we have presented illustrates the new possibilities it provides, in addition to the advantages presented earlier.

6.4 SUMMARY AND PERSPECTIVES

In this chapter, we have reviewed several studies on acoustic instruments, and discussed them from the point of view of *understandability* and *expressiveness*. We have identified various limitations of music software, and proposed a design framework based on an informed structural decomposition of acoustic instruments.

We draw upon other approaches within which the dynamic aspects of the mapping strategy are isolated (e.g. Momeni and Henry, 2006), and extend them by providing an adapted software architecture, and various design guidelines grounded in studies on instrument playing and in the concept of "materiality" of computer systems (Bertelsen et al., 2007). We have improved the use of dynamic mappings strategies by introducing the "adaptation layer" of *behavior models*, the concept of *environment*, and a clearer definition of mapping as simple connections between blocks. We have also identified various advantages of using *behavior models* in mapping strategies, drafting an explanation for the success of music software having their own dynamic behavior.

Our software architecture facilitates experimentations within our framework, with the possibility to save *behavior models* and whole configurations that can be loaded at runtime. Its modularity is supported by the fact that *behavior models* provide direct access to their *handles* and *behavioral measures* when loaded. We think that our framework, the associated software architecture and the observations we have provided can help designers engage in "instrument-making" for music software, and find *expressive* and *usable* combinations of input devices, *behavior models* and synthesis engines.

We also identify several requirements for further experimentations and developments within our software architecture.

Visual representations must help performers make sense of the internal behavior of the system by providing various levels of visual feedback. They can illustrate perceptual parameters of the musical result (e.g., loudness, brightness, timbre, rhythm, melody), or display metaphors of the gestures of the player.

The resulting *materiality* must help users structure their action on and perception of the behavior of the system, by allowing them to implicitly understand it as schemes of movement and not just auditory schemes. Thanks to this physical engagement, accompanying gestures might emerge (Delalande, 1988), easing the *understanding* of the audience attending performances with such systems. As our architecture favors *metonymy* and eases the design of metaphors, *mapping through behavior models* can lead to an important feeling of *materiality*. Our main objective is that novice players, experienced musicians and people in the audience *understand* such systems more deeply than they usually do with music software. We have recently started organizing "conference-concerts" with musician Florent Colautti, to interview people in the audience about their *understanding* of the *behavior models* he uses, whose visual representations are shown on a projection screen during his performances [17].

In order to allow an intimate control of sound production processes, some *behavior models* must be highly *expressive* by taking into account various parameters of the input devices, and reacting to fine changes. In fact, the interaction with acoustic instruments that enables an advanced control of articulation and timbre depends on the precision of control. For example, the sound produced by the friction of a bow on violin strings depends on the speed, direction and pressure of the musician's gesture.

Furthermore, interaction with musical instruments introduces a tradeoff between predictability of the outcomes and "loose" determinism. In fact, a gesture performed in the same situation must always lead to an outcome which is perceived as globally identical from the musical point of view, but musical instruments also introduce variability since equivalent gestures rarely lead to the exact same sound. As we have seen, musicians need to reproduce gestures and then introduce variations in their interaction in order to learn how to play an instrument (O'Modhrain and Essl, 2004a). We think that this consideration is an interesting direction for further experimentations with *behavior models*.

We want to define a taxonomy of *behavior models*, which will be grounded in the study of emergent practices in computer music, as well as in the organology of acoustic instruments, described for example by the *Musical Instrument Museums Online Consortium* (Consortium, 2011). They categorize acoustic instruments according to their

17. See a live performance with the *Eclipse* system at vimeo.com/22836792

energy conversion mechanisms (i.e. idiophone, membranophone, chordophone, aerophone), and identify all the playing techniques and *active surfaces* which are used to interact with them. Studying acoustic instruments in this manner can provide clues as to which are good combinations between *active surfaces* and internal mechanisms.

Such a design space for *behavior models* can be studied together with taxonomies of input devices (e.g. Mackinlay et al., 1990; Wanderley et al., 2000; Marshall and Wanderley, 2006; Antle et al., 2009), ways to control sonic events (e.g. Wanderley, 2001; Levitin et al., 2002; Magnusson, 2010), synthesis engines (e.g. Roads, 1996), auditory events (e.g. Delalande et al., 1996) and attributes of music perception (e.g. Susini et al., 2012). For example, taxonomies of input devices and gestures used for the control of musical processes will help build adapted *active surfaces* and choose visual representations enabling a more embodied interaction with music software.

We believe that it is only by combining these studies with our focus on the essence of acoustic instruments and on the *usability* and *expressiveness* of interactive systems, that music software designers can raise to the status instrument-makers.

7

CONTRIBUTIONS AND CONCLUSION

This dissertation was motivated by four main ideas presented in the first three chapters:

- expert activities are stable repositories of *expressive* tasks and artifacts, as well as adapted learning methods allowing embodied *understanding*;
- non-experts are, in fact, experts in common human activities, including embodied sense-making, social interaction and mediation through artifacts;
- in expressive activities, expert practitioners and non-expert perceivers share intersubjective meaning and use common psychological artifacts;
- knowledge, organized in schemes, can adapt to new situations thanks to the mechanisms of assimilation and accommodation.

In order to describe, evaluate and create interactive systems inspired by expert activities, we drew upon Instrumental Interaction (Beaudouin-Lafon, 2000) and the Human Artifact Model (Bødker and Klokmose, 2011). We defined a novel framework for studying the relationship between users' capabilities and the operational level of interaction instruments (ibid.). We described this relationship in terms of *usability* –including *understandability*, *learnability*, *operability* and *attractiveness*– and *expressiveness*.

We have described both *usability* and *expressiveness* from the point of view of theories derived from the concept of embodiment (Merleau-Ponty, 1945) and from Activity Theory (Vygotsky and Cole, 1978). The background provided in the three first chapters of this dissertation allowed us to describe human sense-making and characterize expert activities. First, expertise implies one's ability to organize perception and action to provide instantaneous analysis of the situation, decision and response corresponding to one's goals. Second, experts automatize their actions and internalize their activities. They use technical artifacts as functional organs by developing refined utilization schemes describing what actions the artifact can handle, how to use it, the results they can expect from their actions and the goals

they can attain. They also build complex psychological artifacts as abstractions to mediate the activities they have internalized.

This description of expert skills was used as a reference point to evaluate the *understandability* and *expressiveness* of our systems. We focused on music as an expressive expert activity that is grounded in inherent internal mechanisms of human beings and have an important social dimension. Therefore, non-musicians have an embodied understanding of musical structure and musical expression. We have described musical expertise as an embodied and mediated activity and identified various music-related skills in non-musicians.

In the fourth chapter, we proposed using rhythmic patterns as an input method. We defined a framework for designing *expressive* vocabularies of rhythmic patterns which are adapted to interaction with computers. We validated that novice users are able to efficiently reproduce, memorize and internalize our vocabularies of rhythmic patterns with just an audio feedback.

In the fifth chapter, we studied the design and *learning* of chording gestures on multitouch screens. We provided a framework for designing *expressive* vocabularies of chording gestures. We validated this framework with users' assessment of the *understandability* of chording gestures and comfort of use. We then designed a learning method inspired by chord teaching in music, and validated that all users were able memorize our vocabularies of chording gestures, while users using our learning method showed a deeper internalization of the gestures.

These two interaction techniques were made even more *accessible* by designing gesture recognizers that do not require training and are ready-to-use.

In the sixth chapter, we proposed a design framework and software architecture, based on the functional properties of acoustic instruments, for creating music software inspired by instrument playing. Within that framework, designing music software is easier and closer to instrument-making. We defined some directions, inspired by our review of studies on instrument playing, to create *expressive* and *accountable* music software allowing novice and expert use, and improving the experience of audiences in the context of live performances.

Considering our main objectives, these projects have shown that, in these cases:

- the tacit knowledge novices have about music can be reused for interaction;
- learning methods inspired by music teaching allows a deep internalization of gestures;

– studying the functional properties of musical instruments can help creating *usable* and *expressive* music software.

Our observation of the mnemonics spontaneously built by the participants in our experiments encourages us to define methods to measure skill acquisition, not in terms of temporal performance, but in terms of knowledge organization and internalization of interaction.

We also want to explore the use of other aspects of music for interaction, such as its syntax. In fact, we are used to understanding and producing structures that are much more than just chunks. In music, notes can be combined in various manners, and non-musicians are used to make sense of the hierarchical structure of pitch or rhythm. For example, we could distinguish between actions that happen at the same time and are oriented towards the same goal —such as chords in music— and sequences of simultaneous actions that operate independently on various objects —such as polyphonic melodies. We could also combine rhythmic patterns presented in chapter four and chording gestures presented in chapter five, and explore the various levels of hierarchy that are accessible with both techniques: rhythmic patterns beginning with the same events and chording gestures involving the same fingers can trigger commands which would have been accessible in the same sub-menu. Therefore, rhythmic chording gestures can rapidly give access to a large number of commands organized in several levels of hierarchy.

In light of the debate on the use of metaphors in interaction design presented in the previous chapter, we can enrich our definition of *understandability*. In fact, the metaphors used in HCI focus on reproducing the visual aspects of a situation that is known to the user. Although defined in our framework as a case in which *understandability* is maximized, we think that reusing embodied tacit knowledge —as in our project on Rhythmic Interaction— is much more powerful than reusing visual similarities that introduce functional limitations. Furthermore, we want to explore the idea presented in the previous chapter of using metaphors of movements instead of visual metaphors. We think that interaction instruments favoring the acquisition of tacit knowledge can be more intuitive than designs favoring immediateness.

We can even facilitate the acquisition of tacit knowledge by reusing the music-related knowledge of non-musicians to make users implicitly *understand* how to interact with the system, or to provide them with information about the state of the system during interaction. We have seen that music is a complex language that non-musicians can perceive and understand. For example, we can create a dynamic guide that is not visual —such as Arpege— but auditory, by providing musical feedback. If the user performs appropriate gestures,

the feedback is tonally consistent. And if the user makes an error, the feedback can show inconsistencies in its musical structure.

Another interesting direction, inspired by the study of musical instruments presented in the previous chapter, would be to consider playing techniques for interaction instruments. For example, a non-musician can easily play a note on a violin with pizzicato, while playing with the bow requires more practice, even for playing a single note. However, both playing techniques do not lead to the same levels of *expressiveness*. In the same manner, we can imagine interaction instruments providing different access to their functions to novice and expert users, shown with the *behavior models* presented in the previous section.

To conclude, we argue that interaction designers can take advantage of the richness of expert activities of which non-experts have an implicit knowledge to create *expressive* and *usable* interactive systems. In this dissertation, we have proposed the study of usability as an alternative to the focus on immediacy that characterizes current commercial interactive systems, and we have established links between different research fields to define informed and lasting ways to consider interaction as an embodied experience.

PUBLICATIONS

- Ghomi, E., Faure, G., Huot, S., Chapuis, O., and Beaudouin-Lafon, M. (2012).
Using rhythmic patterns as an input method.
In *Proceedings of the 2012 ACM annual conference on Human Factors in Computing Systems, CHI '12*, pages 1253–1262, New York, NY, USA. ACM.

- Goudard, V., Genevois, H., Doval, B., and Ghomi, E. (2011a).
Dynamic intermediate models for audiographic synthesis.
In *Proceedings of the 8th Sound and Music Computing International Conference (SMC11)*, Università Di Padova.

- Goudard, V., Genevois, H., and Ghomi, E. (2011b).
L'utilisation de modèles intermédiaires dynamiques pour la synthèse audio-graphique.
In *Actes des 16ème Journées d'Informatique Musicale (JIM11)*, Association Française d'Informatique Musicale.

- Bau, O., Ghomi, E., and Mackay, W. E. (2010).
Arpege: Design and Learning of Multi-Finger Chord Gestures.
Technical Report number 1533, LRI, INRIA.

- Ghomi, E., Bau, O., Mackay, W. E., and Huot, S. (2010).
Conception et apprentissage des interactions tactiles: le cas des postures multi-doigts.
In *Actes du 1er forum sur l'interaction tactile et gestuelle (FITG10)*, Université Lille-1.

- Genevois, H. and Ghomi, E. (2010).
Contrôle gestuel du rayonnement acoustique des sons de synthèse.
In *Actes du 10ème Congrès Français d'Acoustique (CFA10)*, Société Française d'Acoustique.

VIDEOS

The video available at http://www.youtube.com/watch?v=pBqpwtw7PTs illustrates the use of rhythmic patterns as an input method.

BIBLIOGRAPHY

(1929). *The RCA Theremin (advertisement)*. Radiola Division (New-York). Radio-Victor Corporation of America.

Agarawala, A. and Balakrishnan, R. (2006). Keepin' it real: pushing the desktop metaphor with physics, piles and the pen. In *Proceedings of the SIGCHI conference on Human Factors in computing systems*, CHI '06, pages 1283–1292, New York, NY, USA. ACM.

Antle, A. N., Corness, G., and Droumeva, M. (2009). Human-computer-intuition? exploring the cognitive basis for intuition in embodied interaction. *International Journal of Arts and Technology*, 2 (3).

Appert, C. and Fekete, J.-D. (2006). Orthozoom scroller: 1d multi-scale navigation. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '06, pages 21–30, New York, NY, USA. ACM.

Appert, C. and Zhai, S. (2009). Using strokes as command shortcuts: cognitive benefits and toolkit support. In *Proc. CHI '09*, pages 2289–2298. ACM.

Arfib, D., C. J.-M. and Kessous, L. (2005). Expressiveness and digital musical instrument design. *Journal of New Music Research*, 34(1):125–136.

Arfib, D., Couturier, J.-M., Kessous, L., and Verfaille, V. (2002). Strategies of mapping between gesture data and synthesis model parameters using perceptual spaces. *Organised Sound*, 7(2):127–144.

Arom, S., Thom, M., Tuckett, B., and Boyd, R. (1991). *African polyphony and polyrhythm: musical structure and methodology*. Cambridge university press.

Bailly, G., Demeure, A., Lecolinet, E., and Nigay, L. (2008). Multitouch menu (mtm). *Proc. IHM '08*.

Bailly, G., Lecolinet, E., and Guiard, Y. (2010). Finger-count & radial-stroke shortcuts: 2 techniques for augmenting linear menus on multi-touch surfaces. In *Proceedings of the 28th international conference on Human factors in computing systems*, CHI '10, pages 591–594, New York, NY, USA. ACM.

Baily, J. (2001). L'interaction homme-instrument. vers une conceptualisation. In d'ethnomusicologie, A., editor, *Cahiers de musiques traditionnelles : Le geste musical*. Georg.

Bannon, L. J. and Bødker, S. (1991). Designing interaction. chapter Beyond the interface: encountering artifacts in use, pages 227–253. Cambridge University Press, New York, NY, USA.

Banovic, N., Li, F. C. Y., Dearman, D., Yatani, K., and Truong, K. N. (2011). Design of unimanual multi-finger pie menu interaction. In *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces*, ITS '11, pages 120–129, New York, NY, USA. ACM.

Bardini, T. (2000). *Bootstrapping: Douglas Engelbart, coevolution, and the origins of personal computing*. Stanford University Press, Stanford, CA, USA.

Battier, M. (1992). Sculpter la transparence. l'écriture, le geste, l'environnement. In *Cahiers de l'IRCAM - recherche et musique - Composition et environnements informatiques*. Paris: IRCAM - Centre Georges Pompidou.

Bau, O. and Mackay, W. E. (2008). Octopocus: a dynamic guide for learning gesture-based command sets. In *Proc. UIST '08*, pages 37–46. ACM.

Baudel, T. and Beaudouin-Lafon, M. (1993). Charade: Remote control of objects using free-hand gestures. *Comm. ACM*, 36:28–35.

Beaudouin-Lafon, M. (1997). Interaction instrumentale: de la manipulation directe à la réalité augmentée. In *IHM'97: Proc. of the Conf. Francophone sur l'Interaction Homme-Machine*.

Beaudouin-Lafon, M. (2000). Instrumental interaction: an interaction model for designing post-wimp user interfaces. In *CHI '00: Proc. of the 2000 SIGCHI conf. on Human factors in computing systems*, pages 446–453, New York, NY, USA. ACM.

Beaudouin-Lafon, M. (2004). Designing interaction, not interfaces. In *AVI '04: Proc. of the 2004 working conf. on Advanced visual interfaces*, pages 15–22, New York, NY, USA. ACM.

Beaudouin-Lafon, M. and Mackay, W. E. (2000). Reification, polymorphism and reuse: three principles for designing visual interfaces. In *AVI '00: Proc. of the 2000 working conf. on Advanced visual interfaces*, pages 102–109, New York, NY, USA. ACM.

Béguin, P. and Rabardel, P. (2001). Designing for instrument-mediated activity. *Scand. J. Inf. Syst.*, 12(1-2):173–190.

Beilock, S. L., Lyons, I. M., Mattarella-Micke, A., Nusbaum, H. C., and Small, S. L. (2008). Sports experience changes the neural processing of action language. In *Proceedings of the National Academy of Sciences of the United States of America*, volume 105, pages 13269–13273. National Academy of Sciences, Washington, DC, USA.

Bencina, R. (2005). The metasurface: applying natural neighbour interpolation to two-to-many mapping. In *NIME '05: Proc. of the 2005 conf. on New interfaces for musical expression*, pages 101–104, Singapore, Singapore. National University of Singapore.

Bertelsen, O. W., Breinbjerg, M., and Pold, S. (2007). Instrumentness for creativity mediation, materiality & metonymy. In *Proceedings of the 6th ACM SIGCHI conference on Creativity & cognition*, C&C '07, pages 233–242, New York, NY, USA. ACM.

Bier, E. A., Stone, M. C., Pier, K., Buxton, W., and DeRose, T. D. (1993). Toolglass and magic lenses: the see-through interface. In *Proceedings of the 20th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '93, pages 73–80, New York, NY, USA. ACM.

Bigand, E. and Poulin-Charronnat, B. (2006). Are we experienced listeners? a review of the musical capacities that do not depend on formal musical training. *Cognition*, 100(1):100–130.

Blackwell, A. F. (2006). The reification of metaphor as a design tool. *ACM Trans. Comput.-Hum. Interact.*, 13(4):490–530.

Blair, C. R. (1959). On computer transcription of manual morse. *JACM*, 6(3):429–442.

Blakemore, S.-J., Bristow, D., Bird, G., Frith, C., and Ward, J. (2005). Somatosensory activations during the observation of touch and a case of vision–touch synaesthesia. *Brain*, 128:1571–1583.

Bødker, S. and Klokmose, C. N. (2011). The human-artifact model. *Human - Computer Interaction (Mahwah)*, 26(4):315–371.

Bökesoy, S. (2005). The cosmos model: An event generation system for synthesizing emergent sonic structures. In *ICMC'05: Proceedings of the 2005 International Computer Music Conference*.

Bokesoy, S. and Pape, G. (2003). Stochos: Software for real-time synthesis of stochastic music. *Comput. Music J.*, 27(3):33–43.

Bowman, W. and Powell, K. (2007). The body in a state of music. In Bresler, L., editor, *International Handbook of Research in Arts Education*, volume 16 of *Springer International Handbooks of Education*, pages 1087–1108. Springer Netherlands.

Bowman, W. D. (2002). Why Do Humans Value Music? In *Philosophy of Music Education Review*, volume 10, pages 55–63. Indiana University Press.

Bragdon, A., Uguray, A., Wigdor, D., Anagnostopoulos, S., Zeleznik, R., and Feman, R. (2010). Gesture play: motivating online gesture

learning with fun, positive reinforcement and physical metaphors. In *ACM International Conference on Interactive Tabletops and Surfaces*, ITS '10, pages 39–48, New York, NY, USA. ACM.

Brandl, P., Forlines, C., Wigdor, D., Haller, M., and Shen, C. (2008). Combining and measuring the benefits of bimanual pen and direct-touch interaction on horizontal interfaces. In *Proceedings of the working conference on Advanced visual interfaces*, AVI '08, pages 154–161, New York, NY, USA. ACM.

Brochard, R., Touzalin, P., Despres, O., and Dufour, A. (2008). Evidence of beat perception via purely tactile stimulation. *Brain Res.*, 1223:59 – 64.

Bulling, A., Dachselt, R., Duchowski, A., Jacob, R., Stellmach, S., and Sundstedt, V. (2012). Gaze interaction in the post-wimp world. In *Proceedings of the 2012 ACM annual conference extended abstracts on Human Factors in Computing Systems Extended Abstracts*, CHI EA '12, pages 1221–1224, New York, NY, USA. ACM.

Buxton, W. (1986). Chunking and phrasing and the design of human-computer dialogues. In *Proceedings of the IFIP World Computer Congress*, pages 475–480, Dublin, Ireland.

Cadoz, C. (1998). Instrumental gesture and musical composition. In *ICMC'88: Proceedings of the 1988 International Computer Music Conference*, pages 1–12.

Cadoz, C. (1999). Musique, Geste, Technologie. In Genevois, H. and De Vivo, R., editors, *Les nouveaux gestes de la musique*, page 25. Parenthèses.

Cadoz, C., Luciani, A., Florens, J.-L., and Castagné, N. (2003). Acroe - ica: artistic creation and computer interactive multisensory simulation force feedback gesture transducers. In *Proceedings of the 2003 conference on New interfaces for musical expression*, NIME '03, pages 235–246, Singapore, Singapore. National University of Singapore.

Cadoz, C. and Wanderley, M. M. (2000). *Trends in Gestural Control of Music*, chapter Gesture-Music, pages 1–55. IRCAM — Centre Pompidou.

Calvo-Merino, B., Glaser, D., Grèzes, J., Passingham, R., and Haggard, P. (August 2005). Action observation and acquired motor skills: An fmri study with expert dancers. *Cerebral Cortex*, 15(8):1243–1249.

Cance, C. and Genevois, H. (2009). Questionner la notion d'instrument en informatique musicale : analyse des discours sur les pratiques du Méta-Instrument et de la Méta-Mallette. In *JIM'09 - 14èmes Journées d'Informatique Musicale - Actes*, page 133, Grenoble, France. ACROE.

Cance, C., Genevois, H., and Dubois, D. (2009). What is instrumentality in new digital musical devices ? a contribution from cognitive linguistics and psychology. *CoRR November 2009*.

Card, S. and Moran, T. (1986). User technology–from pointing to pondering. In *Proceedings of the ACM Conference on The history of personal workstations*, HPW '86, pages 183–198, New York, NY, USA. ACM.

Cassell, J. (1998). A Framework For Gesture Generation And Interpretation. In Cipolla, R. and Pentland, A., editors, *Computer Vision in Human-Machine Interaction*. Cambridge University Press, Cambridge, UK.

Castagne, N. and Cadoz, C. (2003). 10 criteria for evaluating physical modelling schemes. In *DAFX'03: Proc. of the 2003 conf. on Digital Audio Effects*.

Chen, S.-C., Chien, C.-Y., Chang, W.-M., and Lin, S.-W. (2008). A new assistive communication system for the serious disabled. In *Proc. iCREATe '08*, pages 59–64. START Centre.

Chomsky, N. (1965). *Aspects of the Theory of Syntax*. The MIT Press Paperback Series. M.I.T. Press.

Clark, H. (1996). *Using Language*. Cambridge University Press.

Clarke, E. (1999). Rhythm and timing in music. *The Psychology of Music*, 2:473–500.

Clarke, E. F. (1988). Generative principles in music performance. In Sloboda, J. A., editor, *Generative Processes in Music*, pages 129–178. Oxford University Press.

Committee on Developments in the Science of Learning (2000). *How People Learn: Brain, Mind, Experience, and School: Expanded Edition*. The National Academies Press.

Consortium, M. (2011). Revision of the hornbostel-sachs classification of musical instruments.

Craik, F. I. and Lockhart, R. S. (1972). Levels of processing: A framework for memory research. *J. Verbal Learning and Verbal Behavior*, 11(6):671–684.

Cross, I. (1999). Is music the most important thing we ever did ? music, development and evolution. In Yi, S. W., editor, *Music, Mind and Science*. Seoul: Seoul National University Press.

Crossan, A. and Murray-Smith, R. (2006). Rhythmic interaction for song filtering on a mobile device. In *Proc. HAID '06*, pages 45–55. Springer.

Csikszentmihalyi, M. (1991). *Flow: The Psychology of Optimal Experience*. Harper Perennial.

Dahl, S. and Friberg, A. (2007). Visual perception of expressiveness in musicians' body movements. *Music perception*, 24(5):433–454. QC 20101004. Uppdaterad från submitted till published (20101004).

Dahlstedt, P. (2009). Dynamic mapping strategies for expressive synthesis performance and improvisation. In *CMMR'08: Proc. of the 5th International Symposium on Computer Music Modeling and Retrieval*, pages 227–242, Copenhagen, Denmark. Springer-Verlag.

Davidson, L. and Scripp, L. (1988). Young children's musical representations: windows on music cognition. In Sloboda, J. A., editor, *Generative Processes in Music*, pages 129–178. Oxford University Press.

Davidson, L. and Welsh, P. (1988). From collections to structure: the developmental path of tonal thinking. In Sloboda, J. A., editor, *Generative Processes in Music*, pages 129–178. Oxford University Press.

Davis, C. (2002). *Statistical Methods for the Analysis of Repeated Measurements*. Springer Texts in Statistics. Springer.

De Jaegher, H. (2010). Enaction versus representation: an opinion piece. In T. Fuchs, H. S. . P. H., editor, *The Embodied Self: Dimensions, Coherence and Disorders*. Stuttgart: Schattauer.

De Jaegher, H. and Froese, T. (2009). On the role of social interaction in individual agency. *Adaptive Behavior - Animals, Animats, Software Agents, Robots, Adaptive Systems*, 17(5):444–460.

De Laubier, S. and Goudard, V. (2007). Puce muse - la métamallette. In Lyon, editor, *Proceedings of Journée d'Informatique Musicale (JIM07)*.

Delalande, F. (1988). La gestique de gould. In Guertin, G., editor, *Glenn Gould Pluriel*, pages 85–111. Louise Courteau éditrice, Montréal.

Delalande, F., Formosa, M., and Frémiot, M. (1996). *Les Unités sémiotiques temporelles: éléments nouveaux d'analyse musicale*. Documents Musurgia. Laboratoire musique et informatique de Marseille.

Djajadiningrat, T., Matthews, B., and Stienstra, M. (2007). Easy doesn't do it: skill and expression in tangible aesthetics. *Personal Ubiquitous Comput.*, 11(8):657–676.

Dourish, P. (2001). *Where the action is: the foundations of embodied interaction*. MIT Press, Cambridge, MA, USA.

Dreyfus, H. (1996). The Current Relevance of Merleau-Ponty's Phenomenology of Embodiment. *The Electronic Journal of Analytic Philosophy*, 4:1–16.

Dufourt, H. (1995). *Les cahiers de l'IRCAM, Les Instruments*, chapter L'instrument philosophe. IRCAM - Centre Pompidou.

Dyson, C. (2009). The "authentic dancer" as a tool for audience engagement. In Stock, C., editor, *Dance Dialogues: Conversations across cultures, artforms and practices, Proceedings of the 2008 World Dance Alliance Global Summit*, Brisbane. QUT Creative Industries and Ausdance.

Engelbart, D. C. (1962). Augmenting human intellect: A conceptual framework.

Engelbart, D. C. (1973). Design considerations for knowledge workshop terminals. In *Proceedings of the June 4-8, 1973, national computer conference and exposition*, AFIPS '73, pages 221–227, New York, NY, USA. ACM.

Ericsson, K., Charness, N., Feltovich, P., and Hoffman, R. (2006). *The Cambridge Handbook of Expertise and Expert Performance*. Cambridge Handbooks in Psychology. Cambridge University Press.

Ericsson, K. A., Prietula, M. J., and Cokely, E. T. (2007). The making of an expert. *Harvard business review*, 85(7-8).

Faure, G., Chapuis, O., and Roussel, N. (2009). Power tools for copying and moving: useful stuff for your desktop. In *Proc. CHI '09*, pages 1675–1678. ACM.

Fekete, J.-D., Elmqvist, N., and Guiard, Y. (2009). Motion-pointing: target selection using elliptical motions. In *Proc. CHI '09*, pages 289–298. ACM.

Fels, S. (2004). Designing intimate experiences. In *2004 International Conference on Intelligent User Interfaces*, pages 2–3. Invited Keynote.

Fels, S., Gadd, A., and Mulder, A. (2002). Mapping transparency through metaphor: towards more expressive musical instruments. *Org. Sound*, 7(2):109–126.

FingerWorks (2001). igesture products. http://en.wikipedia.org/wiki/FingerWorks.

Fraisse, P. (1982). Rhythm and tempo. In *The Psychology of Music*, pages 149–180. Academic Press.

Freeman, D., Benko, H., Morris, M. R., and Wigdor, D. (2009). Shadowguides: Visualizations for in-situ learning of multi-touch and whole-hand gestures. *Proc. ITS'09*.

Fuchs, T. and De Jaegher, H. (2009). Enactive intersubjectivity: Participatory sense-making and mutual incorporation. *Phenomenology and the Cognitive Sciences*, 8:465–486. 10.1007/s11097-009-9136-4.

Gallagher, S. and Hutto, D. D. (2006). Understanding others through primary interaction and narrative practice. In Sinha, Itkonen, Zlatev, and Racine, editors, *The Shared Mind: Perspectives on Intersubjectivity*. Amsterdam: John Benjamins.

Geiger, G., Alber, N., Jordà, S., and Alonso, M. (2010). The reactable: A collaborative musical instrument for playing and understanding music. *Her&Mus. Heritage & Museography*, pages 36–43.

Genevois, H. and Ghomi, E. (2010). Contrôle gestuel du rayonnement acoustique des sons de synthèse. In *Actes du 10ème Congrès Français d'Acoustique (CFA10)*. Société Française d'Acoustique.

Ghomi, E., Bau, O., Mackay, W. E., and Huot, S. (2010). Conception et apprentissage des interactions tactiles: le cas des postures multi-doigts. In *Actes du 1er forum sur l'interaction tactile et gestuelle (FITG10)*. Université Lille-1.

Ghomi, E., Faure, G., Huot, S., Chapuis, O., and Beaudouin-Lafon, M. (2012). Using rhythmic patterns as an input method. In *Proceedings of the 2012 ACM annual conference on Human Factors in Computing Systems*, CHI '12, pages 1253–1262, New York, NY, USA. ACM.

Gibson, J. (1966). *The Senses Considered as Perceptual Systems*.

Gill, S. P. (2007). Entrainment and musicality in the human system interface. *AI and society*, 21(4):567–605.

Glass, L. (2001). Synchronization and rhythmic processes in physiology. *Nature*, 410(6825):277–284.

Goormaghtigh, G. (2001). Note sur le jeu du qin. In d'ethnomusicologie, A., editor, *Cahiers de musiques traditionnelles : Le geste musical*. Georg.

Goudard, V., Genevois, H., Doval, B., and Ghomi, E. (2011a). Dynamic intermediate models for audiographic synthesis. In *Proceedings of the 8th Sound and Music Computing Conference (SMC11)*. Università Di Padova.

Goudard, V., Genevois, H., and Ghomi, E. (2011b). L'utilisation de modèles intermédiaires dynamiques pour la synthèse audiographique. In *Actes des 16ème Journées d'Informatique Musicale (JIM11)*. Association Française d'Informatique Musicale.

Goudeseune, C. (2002). Interpolated mappings for musical instruments. *Organised Sound*, 7(2):85–96.

Greenbaum, J. and Kyng, M. (1991). *Design at Work: Cooperative Design of Computer Systems*. Taylor & Francis.

Grossman, T., Dragicevic, P., and Balakrishnan, R. (2007). Strategies for accelerating on-line learning of hotkeys. In *Proc. CHI '07*, pages 1591–1600. ACM.

Gruson, L. M. (1988). Rehearsal skill and musical competence: does practice make perfect? In Sloboda, J. A., editor, *Generative Processes in Music*, pages 129–178. Oxford University Press.

Harrison, C. and Hudson, S. E. (2008). Scratch input: creating large, inexpensive, unpowered and mobile finger input surfaces. In *Proceedings of the 21st annual ACM symposium on User interface software and technology*, UIST '08, pages 205–208, New York, NY, USA. ACM.

Harrison, S., Tatar, D., and Sengers, P. (2007). The three paradigms of hci. In *alt.chi*.

Haury, J. (09). La pianotechnie ou notage des partitions musicales pour une interprétation immédiate sur le métapiano. In *JIM 09: Journées d'Informatique Musicale*.

Heidegger, M. (1927). *On time and being*. Harper torchbooks. Harper & Row.

Henry, C. (2004). Pmpd: Physical modelling for pure data. In *ICMC'04: Proceedings of the 2004 International Computer Music Conference*.

Hinckley, K., Baudisch, P., Ramos, G., and Guimbretiere, F. (2005). Design and analysis of delimiters for selection-action pen gesture phrases in scriboli. In *Proc. CHI '05*, pages 451–460. ACM.

Hinckley, K., Guimbretiere, F., Agrawala, M., Apitz, G., and Chen, N. (2006). Phrasing techniques for multi-stroke selection gestures. In *Proceedings of Graphics Interface 2006*, GI '06, pages 147–154, Toronto, Ont., Canada, Canada. Canadian Information Processing Society.

Hunt, A. and Kirk, R. (2000). *Mapping Strategies for Musical Performance*. IRCAM - Centre Pompidou.

Hunt, A. and Wanderley, M. M. (2002). Mapping performer parameters to synthesis engines. *Org. Sound*, 7(2):97–108.

Hunt, A., Wanderley, M. M., and Kirk, R. (2000). Towards a model for instrumental mapping in expert musical interaction. In *ICMC'oo: Proceedings of the 2000 International Computer Music Conference*.

Hunt, A., Wanderley, M. M., and Paradis, M. (2002). The importance of parameter mapping in electronic instrument design. In *NIME '02: Proc. of the 2002 conf. on New interfaces for musical expression*, pages 1–6, Singapore, Singapore. National University of Singapore.

ISO/IEC (2001). *ISO/IEC 9126. Software engineering – Product quality*. ISO/IEC.

Jakobson, R. and Halle, M. (1956). Two aspects of language and two types of aphasic disturbances. In *Fundamentals of language*, Janua linguarum: Series minor. Mouton, The Hague.

Jaques-Dalcroze, É., Rothwell, F., and Cox, C. (1930). *Eurhythmics, art and education*. Barnes.

Jola, C., Ehrenberg, S., and Reynolds, D. (2012). The experience of watching dance: phenomenological,äìneuroscience duets. *Phenomenology and the Cognitive Sciences*, 11:17–37.

Jordà, S. (2004). Digital instruments and players: part i — efficiency and apprenticeship. In *Proceedings of the 2004 conference on New interfaces for musical expression*, NIME '04, pages 59–63, Singapore, Singapore. National University of Singapore.

Jordà, S. (2005). *Digital Lutherie: Crafting musical computers for new musics' performance and improvisation*. PhD thesis, Universitat Pompeu Fabra.

Jylhä, A., Ekman, I., Erkut, C., and Tahiroğlu, K. (2011). Design and evaluation of human-computer rhythmic interaction in a tutoring system. *Comput. Music J.*, 35(2):36–48.

Kaptelinin, V. and Nardi, B. (2012). *Activity Theory in HCI: Fundamentals and Reflections*. Synthesis Lectures on Human-Centered Informatics Synthesis Lectures on Human-Centered Informatics. Morgan and Claypool.

Kirsh, D. (2009). Explaining artifact evolution. In Evolution. In Malafouris, L. and Renfrew, C., editors, *The Cognitive Life of Things: Recasting the boundaries of the mind*, page 147. Cambridge Press.

Kululuka, A. A. (2001). Du fait gestuel à l'empreinte sonore. In d'ethnomusicologie, A., editor, *Cahiers de musiques traditionnelles : Le geste musical*. Georg.

Kung, P., Küser, D., Schroeder, C., DeRose, T., Greenberg, D., and Kin, K. (2012). An augmented multi-touch system using hand and finger identification. In *CHI '12 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '12, pages 1431–1432, New York, NY, USA. ACM.

Kurtenbach, G. and Buxton, W. (1994). User learning and performance with marking menus. In *Proc. CHI '94*, pages 258–264. ACM.

Kurtenbach, G. P. (1993). *The design and evaluation of marking menus*. PhD thesis.

Kvifte, T. and Jensenius, A. R. (2006). Towards a coherent terminology and model of instrument description and design. In *NIME '06: Proc. of the 2006 conf. on New interfaces for musical expression*, pages 220–225, Paris, France, France. IRCAM - Centre Pompidou.

Lahav, A., Boulanger, A., Schlaug, G., and Saltzman, E. (2005). The power of listening: auditory-motor interactions in musical training. *Annals of the New York Academy of Sciences*, 1060:189–94.

Lakoff, G. and Johnson, M. (1980). *Metaphors we Live by*. University of Chicago Press, Chicago.

Lang, C. E. and Schieber, M. H. (2004). Human finger independence: limitations due to passive mechanical coupling versus active neuromuscular control. *Journal of neurophysiology*, 92(5):2802–2810.

Lantz, V. and Murray-Smith, R. (2004). Rhythmic interaction with a mobile device. In *Proc. NordiCHI '04*, pages 97–100. ACM.

Large, E. W. (2001). Periodicity, pattern formation, and metric structure. *J. New Music Research*, 30(2):173 – 185.

Larkin, O. (2007). Int.lib - a graphical preset interpolator for max msp. In *ICMC'07: Proc. of the 2007 International Computer Music Conf.*

Lee, J. and Kunii, T. L. (1995). Model-based analysis of hand posture. *IEEE Computer Graphics and Applications*, 15(5):77–86.

Leman, M. (2007). *Embodied Music Cognition and Mediation Technology*. The MIT Press.

Leontyev, A. (1981). *Problems of the Development of the Mind*. Imported Publications, Incorporated.

Lepinski, G. J., Grossman, T., and Fitzmaurice, G. (2010). The design and evaluation of multitouch marking menus. In *Proceedings of the 28th international conference on Human factors in computing systems*, CHI '10, pages 2233–2242, New York, NY, USA. ACM.

Lerdahl, F. (1988). Cognitive constraints on compositional systems. In Sloboda, J. A., editor, *Generative Processes in Music*, pages 129–178. Oxford University Press.

Levitin, D. J., McAdams, S., and Adams, R. L. (2002). Control parameters for musical instruments: a foundation for new mappings of gesture to sound. *Organised Sound*, 7(2):171–189.

Lin, J., Wu, Y., and Huang, T. S. (2000). Modeling the constraints of human hand motion. In *Proceedings of the Workshop on Human Motion (HUMO'00)*, HUMO '00, pages 121–, Washington, DC, USA. IEEE Computer Society.

Luque, S. (2009). The stochastic synthesis of iannis xenakis. *Leonardo Music Journal*, 19.1 ():(1):77–84.

MacDougall, H. G. and Moore, S. T. (2005). Marching to the beat of the same drummer: the spontaneous tempo of human locomotion. *J. Applied Physiology*, 99(3):1164–1173.

Mackay, W. E. (1990). *Users and Customizable Software: A Co-adaptive Phenomenon*. PhD thesis, Massachusetts Institute of Technology, Sloan School of Management.

Mackinlay, J., Card, S. K., and Robertson, G. G. (1990). A semantic analysis of the design space of input devices. *Hum.-Comput. Interact.*, 5(2):145–190.

Magnusson, T. (2010). An Epistemic Dimension Space for Musical Devices. In *Proceedings of the International Conference on New Interfaces for Musical Expression*, pages 43–46.

Malacria, S., Lecolinet, E., and Guiard, Y. (2010). Clutch-free panning and integrated pan-zoom control on touch-sensitive surfaces: the cyclostar approach. In *Proc. CHI '10*, pages 2615–2624. ACM.

Malloch, S. (2005). Why do we like to dance and sing? In Grove, R., Stevens, C., and McKechnie, S., editors, *Thinking in Four Dimensions: Creativity and Cognition in Contemporary Dance*. Melbourne University Press.

Malloch, S. and Trevarthen, C. (2009). *Communicative musicality: exploring the basis of human companionship*. Oxford University Press.

Manuel, R. (1960). *Histoire de la Musique*. Encyclopédie de la Pléiade, Gallimard, Paris, France.

Marshall, M. T. and Wanderley, M. M. (2006). Evaluation of sensors as input devices for computer music interfaces. In *Proceedings of the Third international conference on Computer Music Modeling and Retrieval*, CMMR'05, pages 130–139, Berlin, Heidelberg. Springer-Verlag.

Maruyama, Y., Takegawa, Y., Terada, T., and Tsukamoto, M. (2010). UnitInstrument : Easy Configurable Musical Instruments. In *Proceedings of the International Conference on New Interfaces for Musical Expression*, pages 7–12.

Maturana, H. and Varela, F. (1992). *The tree of knowledge: the biological roots of human understanding*. Shambhala.

Maury, S., Athénes, S., and Chatty, S. (1999). Rhythmic menus: toward interaction based on rhythm. In *Proc. CHI '99 EA*, pages 254–255. ACM.

Meltzoff, A. N. (2002). *Elements of a developmental theory of imitation*, pages 19–41. Cambridge University Press, Cambridge, MA, USA.

Merleau-Ponty, M. (1945). *Phénoménologie de la perception: thèse pour le doctorat ès-lettres présentée à la Faculté des lettres de l'Université de Paris*. Bibliothèque des idées. NRF.

Miller, F. P., Vandome, A. F., and McBrewster, J. (2009). *Keyboard Layout: Keyboard (computing), Typewriter, Alphanumeric keyboard, QWERTY, Portuguese alphabet, QWERTZ, AZERTY, Dvorak Simplified Keyboard, Chorded keyboard, Arabic keyboard, Hebrew keyboard*. Alpha Press.

Moelants, D. (2002). Preferred tempo reconsidered. In *Proc. ICMPC '02*, pages 580–583. AMPS.

Momeni, A. and Henry, C. (2006). Dynamic independent mapping layers for concurrent control of audio and video synthesis. *Comput. Music J.*, 30(1):49–66.

Morganti, F. (2008). What intersubjectivity affords: Paving the way for a dialogue between cognitive science, social cognition and neuroscience. In Morganti, F., Carassa, A., and G., R., editors, *Enacting Intersubjectivity: A Cognitive and Social Perspective on the Study of Interactions*. Amsterdam, IOS Press.

Morris, M. R., Wobbrock, J. O., and Wilson, A. D. (2010). Understanding users' preferences for surface gestures. In *Proceedings of Graphics Interface 2010*, GI '10, pages 261–268, Toronto, Ont., Canada, Canada. Canadian Information Processing Society.

Norman, D. (2010a). *Living with Complexity*. MIT Press.

Norman, D. A. (1993). *Things that make us smart: Defending human attributes in the age of the machine*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.

Norman, D. A. (2010b). Natural user interfaces are not natural. *Interactions*, 17(3):6–10.

of Education, N. S. W. D. and Training (2009). *A Guide To Shadowing*. Commonwealth of Australia.

O'Modhrain, S. and Essl, G. (2004a). Enaction in the context of musical performance. In *Virtual Workshop of Enactive Network*.

O'Modhrain, S. and Essl, G. (2004b). Enaction in the context of musical performance. In *Enactive Virtual Workshop*.

Paine, G. (2009). Towards unified design guidelines for new interfaces for musical expression. *Org. Sound*, 14(2):142–155.

Palmer, C. (1997). Music performance. *Annual Review of Psychology*, 48(1):115–138.

Palmer, C. and Drake, C. (1997). Monitoring and planning capacities in the acquisition of music performance skills. *Canadian Journal of Experimental Psychology*, 51.

Palmer, C. and Jungers, M. K. (2003). Music cognition. In *Lynn Nadel: Encyclopedia of Cognitive Science*, volume 3, pages 155–158. London: Nature Publishing Group.

Palmer, C. and Meyer, R. K. (2000). Conceptual and motor learning in music performance. *Psychological Science*, 11:63–68.

Patel, A. D. (2003). Language, music, syntax and the brain. In *Nature Neuroscience*, volume 6, pages 674–681. Nature Publishing Group.

Pelinski, R. (2005). Embodiment and musical experience. In sociedad de etnomusicología, editor, *Transcultural Music Review*, volume 9.

Petitto, L. A., Holowka, S., Sergio, L. E., Levy, B., and Ostry, D. J. (2004). Baby hands that move to the rhythm of language: hearing babies acquiring sign languages babble silently on the hands. *Cognition*, 93(1):43 – 73.

Piaget, J. (1973). *To understand is to invent: the future of education*. Grossman Publishers.

Pirhonen, A. (2005). To simulate or to stimulate? in search of the power of metaphor in design. In Pirhonen, A., Saariluoma, P., Isomäki, H., and Roast, C., editors, *Future Interaction Design*, pages 105–123. Springer London.

Pirrò, D. and Eckel, G. (2011). Physical Modelling Enabling Enaction: an Example. In *Proceedings of the International Conference on New Interfaces for Musical Expression*, pages 461–464.

Polanyi, M. (1966). *The Tacit Dimension*. Number p. 762 in Terry lectures, Yale University. Doubleday.

Potter, D. D., Fenwick, M., Abecasis, D., and Brochard, R. (2009). Perceiving rhythm where none exists: Event-related potential (erp) correlates of subjective accenting. *Cortex*, 45(1):103–109.

Rabardel, P. (1995). *Les hommes et les technologies: approche cognitive des instruments contemporains*. Collection U.: Série Psychologie. A. Colin.

Rammsayer, T. and Lima, S. (1991). Duration discrimination of filled and empty auditory intervals: Cognitive and perceptual factors. *Perception and Psychophysics*, 50:565–574.

Rasamimanana, N., Bevilacqua, F., Schnell, N., Guedy, F., Flety, E., Maestracci, C., Zamborlin, B., Frechin, J.-L., and Petrevski, U. (2011). Modular musical objects towards embodied control of digital music. In *Proceedings of the fifth international conference on Tangible, embedded, and embodied interaction*, TEI '11, pages 9–12, New York, NY, USA. ACM.

Raskin, J. (2000). *The humane interface: new directions for designing interactive systems*. ACM Press/Addison-Wesley Publishing Co., New York, NY, USA.

Rekimoto, J., Ishizawa, T., Schwesig, C., and Oba, H. (2003). Presense: interaction techniques for finger sensing input devices. *Proc. UIST '03*.

Repp, B. H. (2006). Rate limits in sensorimotor synchronization. *Advances in Cognitive Psychology*, 2(2):163–181.

Reynolds, C. W. (1987). Flocks, herds and schools: A distributed behavioral model. In *Proceedings of the 14th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '87, pages 25–34, New York, NY, USA. ACM.

Rizzolatti, G. and Craighero, L. (2004). The mirror-neuron system. *Annual Review of Neuroscience*, 27:169–92.

Roads, C. (1996). *The Computer Music Tutorial*. MIT Press.

Robinson, S., Rajput, N., Jones, M., Jain, A., Sahay, S., and Nanavati, A. (2011). Tapback: towards richer mobile interfaces in impoverished contexts. In *Proc. CHI '11*, pages 2733–2736. ACM.

Rochat, P. and Passos-Ferreira, C. (2009). Three levels of intersubjectivity in early development. In A. Carassa, F. M. and Riva, G., editors, *Enacting Intersubjectivity: Paving the way for a dialogue between Cognitive Sciences*. Università Svizzera Italiana.

Rodger, M., Issartel, J., and O'Modhrain, S. (2007). Performer as perceiver: perceiver as performer. In *Proc. of International Conference on Enactive Interfaces*.

Rovan, J. B., Wanderley, M. M., Dubnov, S., and Depalle, P. (1997). Instrumental gestural mapping strategies as expressivity determinants in computer music performance. *AIMI'97: Proc. of the 1997 AIMI Int. Workshop Kansei - The Technology of Emotion*, pages 68–73.

Sacks, O. (2008). *Musicophilia: Tales of Music and the Brain*, volume 1. Vintage Books, New York, USA, 2 edition.

Scarr, J., Cockburn, A., Gutwin, C., and Quinn, P. (2011). Dips and ceilings: understanding and supporting transitions to expertise in

user interfaces. In *Proceedings of the 2011 annual conference on Human factors in computing systems*, CHI '11, pages 2741–2750, New York, NY, USA. ACM.

Schaeffer, P. (1966). *Traité des objets musicaux*. Collection Pierres vives. Éditions du Seuil.

Schirato, T. (2007). *Understanding Sports Culture*. Understanding Contemporary Culture series. SAGE Publications.

Schutz, A., Walsh, G., and Lehnert, F. (1932). *Phenomenology of the Social World*. Northwestern University studies in phenomenology & existential philosophy. Northwestern University Press.

Scott, J., Dearman, D., Yatani, K., and Truong, K. N. (2010). Sensing foot gestures from the pocket. In *Proc. UIST '10*, pages 199–208. ACM.

Shneiderman, B. (1983). Direct manipulation: A step beyond programming languages. *Computer*, 16(8):57–69.

Sloboda, J. A., Davidson, J. W., Howe, M. J. A., and Moore, D. G. (1996). The role of practice in the development of performing musicians. *British Journal of Psychology*, 87(2):287–309.

Smith, J. O. (2010). *Physical Audio Signal Processing: For Virtual Musical Instruments and Audio Effects*. W3K Publishing.

Song, H., Benko, H., Guimbretiere, F., Izadi, S., Cao, X., and Hinckley, K. (2011). Grips and gestures on a multi-touch pen. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, pages 1323–1332, New York, NY, USA. ACM.

Spain, M. and Polfreman, R. (2001). Interpolator: a two-dimensional graphical interpolation system for the simultaneous control of digital signal processing parameters. *Organised Sound*, 6(2):147–151.

Suchman, L. (1987). *Plans and Situated Actions: The Problem of Human-Machine Communication*. Learning in doing. Cambridge University Press.

Sundberg, J. (1988). Computer synthesis of music performance. In Sloboda, J. A., editor, *Generative Processes in Music*, pages 129–178. Oxford University Press.

Susini, P., Lemaitre, G., and McAdams, S. (2012). Psychological measurement for sound description and evaluation. In Berglund, B., Rossi, G., Townsend, J., and Pendrill, L., editors, *Measurements with Persons*, Scientific Psychology Series. Taylor & Francis.

Swanson, R. and Holton, E. (2001). *Foundations of Human Resource Development*. Berrett-koehler Series. Berrett-Koehler Publishers, Incorporated.

Szentgyorgyi, C. and Lank, E. (2007). Five-key text input using rhythmic mappings. In *Proc. ICMI '07*, pages 118–121. ACM.

Teruggi, D. (2007). Technology and musique concrète: the technical developments of the groupe de recherches musicales and their implication in musical composition. *Organised Sound*, 12:213–23.

Thompson, W. F., Dalla Bella, S., and Keller, P. E. (2006). Music performance. In Faculty of Psychology, U. o. F., in Warsaw, M., and Press&IT, V., editors, *Advances in Cognitive Psychology*, volume 2, pages 99–102.

Thompson, W. F., Graham, P. W., and Russo, F. A. (2005). Seeing music performance: Visual influences on perception and experience. *Semiotica: Journal of the International Association for Semiotic Studies*, 2005(156):203–227.

Tillmann, B., Bharucha, J. J., and Bigand, E. (2001). Implicit learning of regularities in western tonal music by self-organization. In Sougné, R. F. . J., editor, *Evolution, Learning, and Development*, volume Proceedings of the Sixth Neural Computation and Psychology Workshop, pages 175–184. London: Springer.

Trevarthen, C. (1979). Communication and Cooperation in Early Infancy: A Description of Primary Intersubjectivity. In Bullowa, M., editor, *Before Speech: The Beginning of Interpersonal Communication*. Cambridge University Press, Cambridge, UK.

Trevarthen, C. (2000). Musicality and the intrinsic motive pulse: evidence from human psychobiology and infant communication. *Journal of the European Society for the Cognitive Sciences of Music*, pages 157–213.

Van Nort, D. and Wanderley, M. M. (2006). Exploring the effect of mapping trajectories on musical performance. *SMC'06: Proc. of the 2006 Sound and Music Computing Conf.*, pages 19–24.

Vanacken, D., Demeure, A., Luyten, K., and Coninx, K. (2008). Ghosts in the interface: Meta-user interface visualizations as guides for multi-touch interaction. *Proc. TABLETOP '08*, pages 81–84.

Varela, F. J., Thompson, E. T., and Rosch, E. (1991). *The Embodied Mind: Cognitive Science and Human Experience*. The MIT Press.

Vogel, D. and Casiez, G. (2012). Hand occlusion on a multi-touch tabletop. In *Proceedings of the 2012 ACM annual conference on Human Factors in Computing Systems*, CHI '12, pages 2307–2316, New York, NY, USA. ACM.

Vygotsky, L. and Cole, M. (1978). *Mind in Society: The Development of Higher Psychological Processes*. Harvard University Press.

Wanderley, M. M. (2001). *Performer-Instrument Interaction: Applications to Gestural Control of Sound Synthesis*. PhD thesis, University Paris VI.

Wanderley, M. M., Viollet, J.-P., Isart, F., and Rodet, X. (2000). On the choice of transducer technologies for specific musical functions. In *Proceedings of the International Computer Music Conference*.

Watson, A. H. D. (2006). What can studying musicians tell us about motor control of the hand? *Journal of Anatomy*, 208(4):527–542.

Wellner, P. (1991). The digitaldesk calculator: tangible manipulation on a desk top display. In *Proceedings of the 4th annual ACM symposium on User interface software and technology*, UIST '91, pages 27–33, New York, NY, USA. ACM.

Wenger, E. (1999). *Communities of Practice: Learning, Meaning, and Identity*. Learning in Doing. Cambridge University Press.

Wessel, D. and Wright, M. (2001). Problems and prospects for intimate musical control of computers. In *NIME '01: Proc. of the 2001 conf. on New interfaces for musical expression*, pages 1–4, Singapore, Singapore. National University of Singapore.

Westeyn, T. and Starner, T. (2004). Recognizing song-based blink patterns: Applications for restricted and universal access. In *Proc. FGR '04*, pages 717–722. IEEE.

Wigdor, D. and Wixon, D. (2011). *Brave NUI World: Designing Natural User Interfaces for Touch and Gesture*. Morgan Kaufmann. Elsevier Science & Technology.

Wilson, A. D., Izadi, S., Hilliges, O., Garcia-Mendoza, A., and Kirk, D. (2008). Bringing physics to the surface. In *Proceedings of the 21st annual ACM symposium on User interface software and technology*, UIST '08, pages 67–76, New York, NY, USA. ACM.

Winkler, T. (1995). Making motion musical: Gestural mapping strategies for interactive computer music. In *ICMC'95: Proc. of the 1995 International Computer Music Conf.*, pages 261–264.

Wobbrock, J. O. (2009). Tapsongs: tapping rhythm-based passwords on a single binary sensor. In *Proc. UIST '09*, pages 93–96. ACM.

Wobbrock, J. O., Morris, M. R., and Wilson, A. D. (2009). User-defined gestures for surface computing. In *Proc. CHI '09*, pages 1083–1092. ACM.

Wu, J. (2000). Accommodating both experts and novices in one interface. *Practical Design Guidelines for Universal Usability Guide, University of Maryland*.

Wu, M. and Balakrishnan, R. (2003). Multi-finger and whole hand gestural interaction techniques for multi-user tabletop displays. In *Proc. UIST '03*, pages 193–202. ACM.