Interfaces for Sketching Musical Compositions

Anna Xambó Sedó

Master Thesis submitted in partial fulfillment of the requirements for the degree:
Màster en Tecnologies de la Informació, la Comunicació i els Mitjans Audiovisuals

Supervisor: Dr. Josep Blat

Departament de Tecnologies de la Informació i les Comunicacions
Universitat Pompeu Fabra, Barcelona, Spain
September 2008

Abstract

Interfaces for sketching musical compositions are commonly mouse-driven applications that carry, if we compare them to the easy-going physical paper-and-pencil interaction, an execution constraint for that user who prefers to purely concentrate on music ideation. In this thesis we propose that those action-driven and gesture-based interfaces closer to the paper affordance are better suited for the creative tasks than point-and-click interfaces. A usability study is conducted comparing both approaches through a set of traditional computer music tools: an editor, a sequencer and a sampler, in order to evaluate if action-driven interfaces support better creative tasks when compared to similar mouse-driven applications. We obtain positive results that confirm that participants prefer gestural-based interfaces for sketching musical compositions, and we prove that these tools are an adequate environment for enhancing creativity, both for computer musicians and non-musicians.

Acknowledgments

First of all, I would like to thank Josep Blat, my supervisor, for supporting my research along this year and giving me the opportunity to both present part of this work to and receive feedback from the *Grup de Tecnologies Interactives* (GTI) of the Universitat Pompeu Fabra.

Also I am truly indebted to Xavier Serra, director of the Music Technology Group (MTG) of the Universitat Pompeu Fabra, for giving me the opportunity to present part of this research at the Summer School of Sound and Music Computing held in Genova this year (SMC 2008) and for letting me to be part of the project *Sons de Barcelona*[1], whose concept has influenced the tr4ck prototype, which allows to cut and mix sounds everywhere.

I wish to specially thank all the professors of this TICMA master program for their patience and enthusiasm and for letting this happen. The prototypes presented in this thesis would not be the same without their guidance. Special thanks to the reacTable team Sergi Jordà, Günter Geiger, Martin Kaltenbrunner and Marcos Alonso for their support in the waveTable prototype, and also special thanks to Bert Bongers for his support in the wiiteBoard prototype. Also, I would like to thank all the members of the MTG and the colleagues of the TICMA master for their support. In addition special thanks to Alexander Refsum Jensenius and Antonio Camurri for their feedback in the SMC 2008.

I would like to mention and thank initiatives like the N810 Maemo program led by Quim Gil from Nokia, a program that has allowed me to both enjoy and develop the tr4ck prototype in this Internet tablet. I wish to thank as well the free software community for developing powerful computer languages such as SuperCollider, Processing, PD and PDa.

[1] http://barcelona.freesound.org

I wish to thank as well the following people for their help, support, comments and suggestions, in alphabetical order: Carlos Alocén, David Barjie, Pep Guasch, Toni de Jaumar, Marc Jarpa, Martin Haro, Katalin Pula and Marta Xambó.

Last, but not least, I wish to thank my parents, who have always encouraged me, and Gerard, who apart from developing together two of the three prototypes presented in this thesis, has patiently supported this long and fruitful master academic year.

Anna Xambó
Barcelona
September 15, 2008

To Gerard

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In the laboratory that I set up twenty years ago, we developed a system whereby anybody at all can compose music by drawing. It is both a tool for composers and acoustics experts and a teaching aid for children, who can learn to think in music without having to be trained in musical notation–learning by direct experience, in other words. This could not have been done without informatics, which yielded possibilities comparable to those brought by the invention of writing: preserving thought in symbols. Here, we can use our machine to preserve musical thought, since we can also store it.

—Iannis Xenakis

Of the computer tools offering musical support there are few devoted only to musical composition and the creative process. We often use high-end programs that eventually may offer this option but that are not specifically designed for that purpose. This approach drives the musician, expert and beginner to choose between two common scenarios: one consists in working directly with the music software, employed as a digital instrument with its own possibilities and limitations which condition thought and creativity. The other separates two stages, sketching the ideas using alternative physical resources versus arranging them using the computer, forcing the musician to perform a conversion compromise from the sketch to the final version. Both strategies carry some drawbacks: while the former is directly subjected to the music software environment of what it is (and it is not) allowed to perform, the second is time consuming.

Literature on the subject presented in this thesis sustains that this lack of computer-support for sketching is in part due to the difficulty to model the creative process, where some creativity studies focused on the music domain propose to understand the interaction as a key factor. In that direction we find that sketching musical compositions is a wide-range activity that may be undertaken in different contexts such as the home studio, the mobile studio or even live performance. Within these different contexts several exploratory activities can be performed like composing a melody by drawing, by recording sounds (in the tradition of humming but also soundscaping) or by changing an existing song. All these possible techniques drive us to ask what the best interface and input device for each case would be. In this thesis we propose that the physical activity determines the appropriate interaction, therefore those digital interfaces that are conceptually closer to their homonymous physical interfaces, say, for instance, drawing on a pen-and-pencil notebook, painting on a brush-and-canvas or sculpting on a human-scale surface, can produce qualitatively better results and with lower cognitive load to the user than those traditional mouse-driven interfaces for musical sketching with textual menus and modal windows. The approach we propose can be implemented using traditional computer music tools such as an editor, a sequencer or a sampler but with an emphasis on both the sketching process and the activity-based interaction.

In order to assess these alternative interfaces, a comparative usability study between pairs of interfaces is done, with participants who do not necessarily have musical background, comparing the accomplishment of composition tasks. The results seem to confirm that activity-driven interfaces are adequate for sketching musical compositions, mainly for non experienced musicians, as well as indicating their degree of usability based on information about learnability, usefulness, experience and performance obtained by analizing the collected data.

1.1 Motivation

Sketching musical compositions is a creative, exploratory and playful activity. It can be done by improvising when playing an instrument with your band or being alone, writing down some ideas that can be sonically represented later on, humming a melody and recording it. Composing music with computers has become so popular that being a musician is not anymore a question of years studied in the academy. During the last years, the available computer tools provide support for each step or the whole process of the musical production, although surprisingly the first stage of composing, namely musical sketching, is little supported.

In recent years, the development of tangible user interfaces (TUIs) and their adequacy to embodied tasks has highlighted that those activities less desktop-centered such as collaborative or creative activities fit well with these systems. Similarly, the popularization of handheld devices has emphasized that entertainment, communication and creative activities suit well these systems. Thus, I have explored the possibilities of these technologies to enhance musical creativity.

1.2 Goals and thesis structure

The goals to be achieved in this thesis can be itemized as follows:

- To summarize the scientific background of creativity support tools for composing music.
- To discuss state-of-the art systems, according to tangible, pen-driven and mobile interaction models.
- To propose several prototypes for sketching musical compositions based on these interaction models.
- To define an evaluation methodology applicable to interfaces for sketching musical compositions.
- To conduct a usability test with trained and untrained musicians.
- To discuss the results considering both quantitative and qualitative data.
- To highlight contributions, open issues and future work.

In order to fulfill these goals, the thesis is based on the following structure: At the beginning there is a scientific background summary of creativity support tools for composing music, which is divided into six sections. The first one outlines the relation between sketching and the creative process and how it is computer-supported, the second section presents sketching within the computer music context, the following section introduces which HCI evaluation methods have been employed in order to assess interfaces of musical expression, and the next three sections explain three interaction models that are close to sketching activities: tangible, pen-driven and mobile interaction.

In the next chapter, three prototypes are presented. First, the tangible tabletop waveTable, an editor which allows to sculpt a digital waveform on a table using a tangible toolkit; second, the wiiteBoard sequencer, a sequencer which enables to draw on a vertical computer screen using a light pen in the form of brush-on-canvas;

and third, the tr4ck recorder, a mobile pocket-size sampler that allows to record up to four tracks and mix them in real time using a stylus on a PDA screen in the form of pen-and-paper.

In the following chapter the evaluation method is presented which consists of a description of the usability study and an overview of the tasks and interfaces to be evaluated. Next, an analysis of the usability study results is outlined according to the quantitative and qualitative data gathered, ending with a discussion about which interfaces are the most useful for sketching musical compositions.

Finally, the conclusions are pointed out, wrapping up contributions, highlighting open issues and future work, and ending with general conclusions.

Chapter 2

Background

2.1 Creativity support tools

In this section we present sketching and the role it plays within the creative process. After that, interactive sketching is described within the context of creativity support tools, establishing correlations between the visual and the music domains. Mental models, metaphors and affordances are introduced as cue factors for designing user-centered interfaces that support creativity.

2.1.1 Understanding the sketching process

Analyzing the literature on creativity related to innovation, discovery and design, B. Shneiderman outlines three main non-excluding schools for defining creative processes: *structuralists* who defend to follow an organized method in order to succeed in being creative (e.g. preparation, incubation, illumination and verification), *inspirationalists* who search for inspiration by producing unusual or unfamiliar situations (e.g. playful exploration, sketching or concept mapping), and *situationalists* who seek connections between creative people and their socio-cultural environment (e.g. influencing relationships, family history or the role of rewards and recognition) (Shneiderman 2007).

During the creative process, one common practice is to create external representations in order to allow individuals to refine their thoughts. Sketching is an inspirationalist activity employed for that purpose. Nakakoji et al. observe that sketching is practiced by disciplines from art, design, engineering and research domains, and provide a broader definition of sketching than hand-drawing on paper using pencil, which includes all those easy, fast, spontaneous and draft representations (Nakakoji et al. 2006). In that sense, we can include recordings, pictures, texts, or anything that can help us illustrate our ideas. For understanding the sketching process, the authors recognize that each domain employs their own process of creation and representation, but they seek a common ground applicable within HCI research that can help in the development of generic tools that support creative processes. Coughlan and Johnson also recognize that for designing creativity support tools there are cross-domain similarities versus other domain-specific aspects to be considered, positioning musical composition as an interesting domain for exploring ideas applicable to other domains (Coughlan and Johnson 2006). Healey and Thiebaut point out that, although music is a more abstract domain than others that are inherently visual, sketching plays a key role in facilitating the development of ideas in musical composition (Healey and Thiebaut 2007). The authors claim a better understanding of the creative process in order to provide tools adequate for this activity. According to Coughlan and Johnson, interaction in creative tasks are cue for understanding the creative process.

2.1.2 Interactive sketching

Schneiderman provides a general picture about computer-based creativity support tools and their influence in the creative process: on one hand, they enable new forms of expression because of their property of extending the individual's abilities to make discoveries; on the other, they facilitate both individual and social creativity due to their property of enabling multidimensional communication across space or time, among others (Shneiderman 2007). The author mentions that these tools are a focus of growing interest in the scientific and engineering domains, enabling to refine research methods for measuring creativity (§2.3).

Traditional computer interfaces using mouse and keyboard with toolbar buttons or menu items are seen as too formal and accurate for creative tasks that require more conceptual design (Landay and Myers 1995, Xu et al. 2002, Adler et al. 2004). In addition, they are difficult to be displayed on handheld devices with small screens and with reduced keyboards, if any (Xu et al. 2002). In general, these interaction issues have led to the development of sketch-based user interfaces using pen-driven interaction, principally employed for design purposes (Landay and Myers 2001, Xu et al. 2002, Adler et al. 2004). The coupling of sketching activity and gesturing with a pen can be explained because of both are two similar modes of informal interaction able to produce rapid, uncertain or ambiguous representations (Landay and Myers 2001). In summary, these interfaces try to maintain the benefits of paper-based sketching with those of electronic tools (§2.5).

The pioneer researcher Ivan Sutherland introduced Sketchpad in 1963, a man-machine graphical communication system where the user is able to point at and interact with the objects displayed on a screen using a light pen, that preceded the mouse. According to Blackwell, the capacity of selecting an object by pointing is one of the discoveries further incorporated in the principles of direct manipulation (Blackwell 2006). Sutherland presents his system as follows (Sutherland 1963):

> ′′ The Sketchpad system uses drawing as a novel communication medium for a computer. The system contains input, output, and computation programs which enable it to interpret information drawn directly on a computer display. It has been used to draw electrical, mechanical, scientific, mathematical, and animated drawings; it is a general purpose system.´´

It is worth to mention how the author describes a general-use and flexible system that recognize graphical elements from user drawings and accept behaviours, features that later on will be common in traditional electronic sketching tools. However, the accuracy achievable with that system is another feature more common in current high-end drawing programs.

Bill Buxton explains how the iterative design is as a continuum dialogue between sketches and prototypes (Buxton 2007). Beside this approach, we find several user interface (UI) design systems such as SILK, a tool for sketching interfaces using an electronic pad and a stylus on a workstation that recognises interface elements drawn by the user (Landay and Myers 1995, Landay and Myers 2001) or DENIM, a tool for sketching web site design where it is possible to create hyperlinks quickly among the web pages (Klemmer 2004). This viewpoint allows us to verify immediately the interaction design, thus accelerating the whole creative process.

In music domain, Healey and Thiebaut run a survey among computer literate composers of contemporary music with the aim of getting an overview of sketching in musical composition (Healey and Thiebaut 2007). They conclude that although composers make significant use of sketching, they generally employ paper-and-pen instead of computer music tools. According to the authors, during this early stage composers are involved in an iterative dialectical process and sketches suit well because of allowing revision and reinterpretation of ideas rapidly. However, the case study exemplifies how the composer is involved in a further step of compromise from the sketch to the software, that probably could be reduced using a tool for interactive sketching. Although there is a range of technologies that allow to edit and mix sounds, the authors claim that there are few computer music tools that support the creative process and propose that supporting sketching would imply supporting composing music. In this thesis we explore this gap between sketching with paper-and-pen and editing or mixing with high-end computer music tools establishing correlations between the paper-and-pen interface and those interaction models that can play a similar role using computers (§2.4, §2.5, §2.6).

2.1.3 Mental models, metaphors and affordances

In order to understand how humans think computer software works, we find in HCI and interaction design literature a distinction between three representational models: the *system* or *implementational model*, which is defined as the actual representation of how a machine or a program works and its code implementation; the *mental* or *conceptual model*, described as the user representation when using a machine or a program; and the *represented* or *designer's model*, where due to the differences between the user's mental model and the complexity of the implementation model, in particular in the case of software applications, there is a layer in-between that represents the computer's functioning in the form of a filtered explanation, an inter-face created by the programmer or designer (Norman 2002, Cooper et al. 2007a) . Generally, the closer is the represented model to the mental model, the easier is for the user to learn and use the program or environment because it better matches how people think and work, and the other way around.

The role of the designer's model is traditionally accomplished through interface metaphors, which are representations or "literary" descriptions that aim at helping the user understand the abstract operations and possibilities of the computer (Blackwell 2006). Blackwell states that, although there are varied range of opinions from coupling graphical user interfaces (GUIs) to metaphors, to those that prefer invisible user interfaces (UIs) and ubiquitous computing systems, there is a body of research that support the historical and functional role of metaphors in UI design. In present days, the challenge for practitioners and researchers is to experiment novel UIs getting rid of the *desktop metaphor*, which for the moment and despite its detractors, might be considered as a successful design tool. According to the author, UI designers should adopt an experimental approach when constructing metaphors using new analogies and correspondences (§2.2.3).

Affordance, or the quality of an object, is a term widely used in HCI. Within this area, it is applied to the human perception of those possible actions to be performed with an object. The association of affordance to the way we interact with an object was introduced by Donald Norman (Norman 2002) who applies the concept to the psychology of our everyday objects, say, those essential properties of an object that communicate how it can be used: knobs to be turned, balls to be bounced or doors to be pulled or pushed are some examples given. According to the author, we first try to derive the functioning of the objects around us avoiding as much as possible reading their instruction manuals. Some electronic devices are designed under metaphors and affordances coming from everyday objects: the PDA's interface, for instance, is similar to the paper-and-pen interface, and both can be used getting rid of any signal (see figure 2.1).

Figure 2.1: Affordances of paper-and-pen interface. Left image affords writing with the pen on the sheet of the notebook paper and right image affords writing with the stencil on the screen of the Nokia N810 PDA.

The three prototypes presented in this thesis employ interface metaphors and affordances that escape from the traditional desktop metaphor with the aim of enriching the creative experience, although they are similarly based on simulating non digital artifacts well-known by users: a tangible toolkit on a table in the waveTable editor prototype, a canvas-and-brush on a computer's screen in the wiiteBoard prototype and a handy multitrack recorder on a PDA's screen in the tr4ck prototype.

All three opt for transparent mapping (§2.2.3).

2.2 Computer music

In this section we talk about those design issues related to sketching musical compositions using computers. Hence, the Miranda's framework of composing electronic music is presented in order to define a theoretical scenario for sketching sound compositions. After that, the notion of interactive music system is introduced and a quick overview of those devoted to musical creativity is done. Finally, some strategies of mapping between actions and sound output are presented in the context of musical sketching.

2.2.1 Composing music with computers

When composing music with computers, there are some compositional approaches to be considered. First, a composition implies to manage musical structures which can vary in granularity. E. R. Miranda describes three levels of abstraction when composing music with computers: the *microscopic level*, the *note level* and the *building-block level* (Miranda 2001) . At the *microscopic level*, the composer works with physical and low-level sound features such as frequencies and amplitudes of the partials of sounds, mainly used in synthesis and sound processing. At the *note level*, the music element is a musical note, a single sound which is considered as a sum of sound attributes such as pitch, duration, dynamics, timbre, and also tempo. At the *building-block level*, composers work with larger musical units in the form of rhythmic, melodic or sampled sound sequences. In this third level, the composer works with high-level sound features and, in accordance with some survey results (Healey and Thiebaut 2007, Coughlan and Johnson 2006), it matches to sketching musical compositions because of letting the musician to represent ideas with less cognitive load than when working with low- or mid-level features of sound.

Similarly, Miranda distinguishes between the *bottom-up* approach which favors the experimentation and discovery producing material that can be reused in the future and the *top-down* approach which emphasizes starting with an overall compositional structure that can be refined during the composition process. Due to sketching nature explained in §2.1.1 and §2.1.2, a system for sketching musical compositions should support the *bottom-up* approach. Paradoxically, those computer-aided composition applications that help composers in organizing ideas tend to support the top-down approach because they are principally focused on high-end musical production (§2.1.2).

Miranda, in addition, explains that during the compositional process there are two types of activities: *conceptual activities* and *writing activities*. While the first is described as defining rules and strategies of how will be the the musical style using any format depending on each composer (sketches, diagrams, math symbols, drawings, texts, etc...), the second, instead, is characterized as representing the musical concepts through decision making (notes, instrumentation, and so forth). In-between it is presented the compositional model, which mediates turning ideas into music and where the computer can be used as an excellent tool for that purpose due to its characteristic feature of modelling tasks (see figure 2.2).

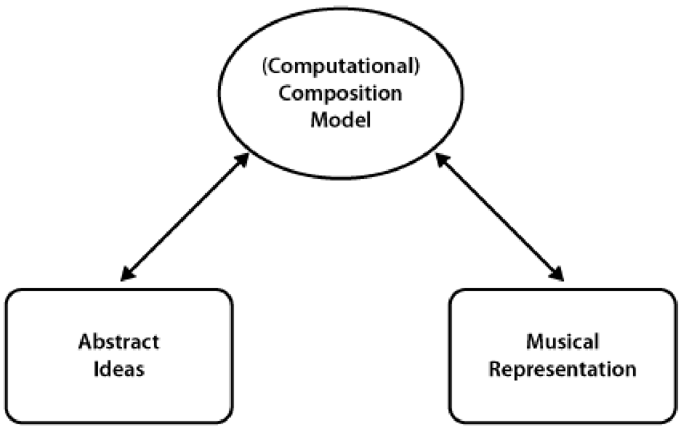

Figure 2.2: Miranda's Composition model.

In summary, according to Miranda's framework an optimal scenario of interfaces for sketching musical compositions should, first, remain in the abstraction boundary of the building-block level in order to facilitate the task of representing musical ideas from a user-centered viewpoint where the designer's model is close to the mental model (§2.1.3); secondly, support a bottom-up approach for composing that can promote the discovery and exploration during the user experience; and ultimately, offer a computational composition model that enables the composer to turn musical ideas into music.

2.2.2 Creativity and interactive music systems

Sergi Jordà exposes how *interactive music system* is a term used for the last two decades with a wide range of meanings that mainly vary according to the developments of computer music history (Jordà 2005). The author proposes a definition of these systems, first describing three common properties:

1. are computer-based,
2. are interactive,
3. generate a musical output at performance time, under the control of one or several performers.

According to Jordà, the *interaction*[1] is possible through a continuous conversation between the performer and the computer system, which has to be interactive enough to have an effect on and modify the performer(s) actions. The possibilities of exploration and discovery produce a combination of features from composition and performance, namely improvisation.

Studies of designing tools to support musical composition have addressed the issue of interaction methods. According to Coughlan and Johnson, computer support to the composition process is equivalent to supporting the creative process in general: there are cycles of ideation and evaluation, thus an important design goal is to connect expression to rapid ideation and evaluation (Coughlan and Johnson 2006). Within this context, there is a clear connection between musical composition support tools and interactive music systems, thus, if not only deals the creative dialogue with elements of composition but also of performance (like being in real time), we find a similar situation than musical improvisation. All three prototypes presented in this thesis, independently of the interaction model employed, exploit this connection where the composing process is a continuous and real-time dialogue with the system, an improvisational dialogue that may nurture the creative process.

2.2.3 Mapping

Mapping is a complex topic that deals with computer systems and the associations established between input as control and output as a response to that input. It directly involves digital music instruments (Jordà 2005, Magnusson 2006, Hunt

[1] *Interaction* defined as the action or influence of people, groups or things on one another (quoted in (Jordà 2005)).

et al. 2002): Playing acoustic instruments is full of subtleties, where the playing affordances are determined by the physical material (Magnusson 2006), and the gestures are coupled with sounds. However, in software the absence of the physical instrument is changed by a mapping or system translation of the musician gestures as the input into the sound produced as the output. According to performers, mapping is the core of how the instrument 'feels' (Jordà 2005). Assuming that an electronic instrument is a combination of an interface and a sound generator, mapping consists in associating input parameters to system parameters (Hunt et al. 2002), where the interface plays an essential role that establishes a semiotic system that determines the performer (Magnusson 2006).

This separation between the input gestures and the output sound invites to discover creatively through new mappings and metaphors (Jordà 2005, Magnusson 2006). According to Gadd and Fels, metaphor (§2.1.3) determines mapping, and the more transparent is the mapping the more expressive will be the instrument (Gadd and Fels 2002). The authors point out that this approach can be accomplished using *common knowledge* metaphors such as the metaphor of rainfall to make understandable the process of granular synthesis or running water to produce musical sounds. But, when users expect a behaviour unaccomplished, metaphor may limit the system that could otherwise invite to explore new relations. Mappings and metaphors influence in how do we approach and learn an instrument. According to Blackwell, learning can be improved with software metaphors based on varied and different elements because invite users to build their own meanings rather than UI with traditional mappings (Blackwell 2006). Current examples of original approaches to musical interfaces, paradoxically screen-based and mouse-driven, are the IXI software project and its computer music applications[2], as well as the editor and player of the Freesound Radio[3].

Mapping is traditionally classified as *one-to-one*, *one-to-many*, and *many-to-one*, by means of the number of input and output parameters associated as independent control dimensions (Jordà 2005). One-to-one is the simplest environment: each physical input parameter (e.g. knobs, sliders, etc.) affects one sound output parameter (e.g. frequency, amplitude, etc.). From a user-centered viewpoint, one-to-one is the easiest mapping that can be understood by users, however it is characterized by being neither expressive nor flexible when compared to the other groups. The Theremin is a paradigm of this one-to-one mapping, that paradoxically allows

[2] http://www.ixi-software.net
[3] http://www.freesound.org/radio

the performer to achieve a high degree of expression: there are two antennas, each one controlling independently frequency and amplitude as you move your hands around[4].

For a better understanding of musical mapping, several authors have integrated mappings into mathematical formulations. The easier case would be a function that modifies all the elements of a set *I* (*Input*), cited in (Jordà 2005):

$$\forall x \in I, \exists y / y = f(x)$$

Mapping between music and the visual sketch is not straightforward. Music is an abstract domain not inherently visual with a wide range of non-spatial dimensions such as pitch, timbre, intensity, tempo or rhythm than the common spatial dimensions of the visual domain (Healey and Thiebaut 2007).

Some systems dare these difficulties and propose their own metaphors and correlations. The Sonic Sketchpad prototype (Coughlan and Johnson 2006) is a tool able to represent, record and share ideas from a traditional viewpoint: the musician plays an standard instrument and records an idea using a foot pedal, being able later on to evaluate it and continue the session. The recordings are added to a free-form interface on a portable tablet PC, where it is possible to add annotations with an stylus, draw or include a picture establishing connections between recordings and sketched information. Music Sketcher, on the other hand, is a tool for composing at the building-block level (§2.2.1) that offers a collection of short musical phrases (*riff blocks*) that can be in turn combined and transformed in order to create a larger musical fragment (Miranda 2001). Currently this concept can be found in most of the commercial digital audio editing and music sequencers such as Ableton Live[5] (§4.3.3), Cubase[6] or Nuendo[7]. The musician can drag and drop from a library to a score with several tracks, and it is possible to apply effects (*modifiers*) in the form of graphical curves drawn by the user, each one affecting one aspect of the music content (e.g. pitch, articulation, loudness, velocity, duration, onset, etc.). This mechanism is based on a one-to-one mapping and it is multi-dimensional given the amount of modifiers at one's disposal.

All the prototypes presented in this thesis are built using one-to-one mapping. The wiiteBoard (§3.2) prototype is two-dimensional, being possible to modify fre-

[4] http://en.wikipedia.org/wiki/Theremin
[5] http://en.wikipedia.org/wiki/Ableton_Live
[6] http://en.wikipedia.org/wiki/Steinberg_Cubase
[7] http://en.wikipedia.org/wiki/Nuendo

quency (pitch) and time (duration) depending on the x- and y-axes of the light pen. The track prototype (§3.3) is two-dimensional, where you can control independently amplitude (volume) and duration (loop length) depending on the status of their respective sliders. Finally, in the waveTable (§3.1) prototype each tangible has its own one-to-one mapping that may depend on the position (x- and y-axes) as well as the angle of the object. The effects, for instance, are mapped as follows: the x-axis determines the position of the envelope, the y-axis the value of the envelope and the rotation the value of the external envelope points.

2.3 HCI evaluation of interactive music systems

In this section we describe how related are the evaluation of interactive music systems to HCI methods, concretely, those approaches that are concerned in general with the assessment of new interfaces for musical expression and those that are concerned in particular with the evaluation of systems for sketching musical compositions. To our knowledge, in the literature the assessment of interfaces for music creation is done conducting a usability study with two phases: data recruitment and data analysis. The methodology is fundamental in order to succeed in the results: that implies the choice of a qualitative or quantitative approach or a balance between both.

2.3.1 Evaluation of new interfaces of musical expression

The evaluation of new interfaces for musical expression is considered a novel field of research: Up to present, a revision of the NIME conference proceedings, international conference concerned with new musical technologies that in fact started in 2001[8] as a CHI conference's workshop (Poupyrev et al. 2001), shows that few of the papers have been devoted to the evaluation of novel instruments using HCI methods (Stowell et al. 2008, Kiefer et al. 2008). The benefits of HCI evaluation may spread from improving design and creativity, defining reusable design patterns to identifying the future instrument influence in both the creative and the technological domains.

On one hand, the research community recognises commonalities between the design of input devices in HCI and the design of controllers for musical expression, pointing out that the latter consists in an specialized field of HCI that is often

[8] http://www.nime.org

submitted to artistic and therefore creative demands (Orio et al. 2001). According to Orio et al., the subtle differences between both invite to start out from the traditional HCI evaluation of comparing devices performance by accomplishing similar tasks (Buxton 1995) but adapted to the musical context. Hence, a framework for assessing musical tasks is proposed by the authors which apart from the classical HCI performance measurement also considers the subjective opinion of the performer. The model states the evaluation of very simple musical tasks as the basic unit of measurement which should take into account *learnability* or the time needed to learn including replication of musical gestures, *explorability* of the range of gestures perceived by the performer, *feature controllability* meaning the perceived accuracy of features when performing musical tasks and, lastly, *timing controllability* in terms of the perceived temporal control by the performer. The suggested musical tasks are the performance of isolated tones, basic musical gestures, simple scales and arpeggios, phrases with different contours, continuous feature modulation, simple rhythms using tones and pre-recorded material or synchronization of musical processes.

On the other hand, some approaches are less focused on the precision of the musical task highlighting the quality of the experience and degree of expressiveness. In addition, they also take into account musical systems that may include random and stochastic elements that in the previous framework are not considered at all. Stowell et al. (Stowell et al. 2008), for instance, employed the qualitative and structured method of *Discourse Analysis*, filtering manually the most important concepts from interviews or written texts: After running semi-structured interviews videotaped around two similar voice-based interfaces for controlling musical systems, the authors proceed with data analysis by transcribing manually the speech in order to identify the world of the subject and get a global picture. This methodology is highly time-consuming and maybe the flexibility sought when searching for free-tagging could be achieved using more automatic and less time-consuming procedures.

Kiefer et al. conducted a comparative usability study that evaluates the usefulness of the Nintendo Wiimote as a musical controller compared to the similar controller Roland HandSonic (Kiefer et al. 2008). The usability study consists in testing the musical capabilities of the devices performing simple musical tasks, all videotaped and tracked through a log file for later analysis of quantitative data. Before the tasks performance, users are allowed to practice for a while, and after each task, participants are interviewed using the *think aloud* method. In conclusion, the authors say that the most intereseting results come from the analysis of interview data rather than from quantitative data and validate the usefulness of the Wiimote as a musical controller, but highlight the need of data about the real-time experience of users using the device. Kiefer et al. affirm that the intersection between HCI evaluation methodology and computer music is a novel field, and claim for research focused on adapting these methodologies in the musical domain. For that purpose, it should to be considered interaction models different than windows, icons, menus and pointers-based (WIMP), and adopt a qualitative and timesaving approach given the organic nature of performing creative tasks.

Although highlighting the user experience, another approach is to assess open-ended musical devices in order to explore further design possibilities. Bau et al. (Bau et al. 2008) apply the HCI participatory design methodology in order to design and test the A20 musical device, a prototype that has audio input/output allowing users to explore music and sound in a tangible interface. The evaluation consisted in first, assessing the perceptual characteristics of the device (sonic and haptic) performing a set of tasks and second, inviting users and designers to imagine new interfaces of the instrument based on several interaction mappings of gesture-based interaction. The modularity of this approach allows to share with users the iterative design of prototypes, that can be principally useful for discovering expected and unexpected functionalities of novel devices.

So far, the papers presented at NIME concerning HCI evaluation methodology applied to music technology keep in general a qualitative and user-centered approach (Bau et al. 2008, Kiefer et al. 2008, Stowell et al. 2008). The interest remains on evaluating the interaction between a musician, composer, sound designer or performer with the instrument, and the majority of criteria evaluated such as usability, expressiveness or aesthetics, among others, is within the user experience domain. Frequently these criteria is evaluated from a comparative viewpoint based on comparing similar musical instruments (Kiefer et al. 2008, Stowell et al. 2008). The choice of gathering quantitative data is little done because assessing musical expression using quantitative methods is described as problematic due to the subjectivity of performing creative tasks, and when employed tends to provide complementary and non-relevant information (Kiefer et al. 2008).

2.3.2 Evaluation of interfaces for musical sketching

Many studies informally evaluate their prototypes of sketching musical compositions based on observation and gathering qualitative data. During the session, after a period of tutorial and practice, the participant is asked to perform, individually or in group, a creative task(s) that will be evaluated (e.g. *compose something you are happy with*) (Coughlan and Johnson 2006). Later on, questions about usefulness are made like how the use of this tool affects the creative process, how useful it is considered and if the participant would used it in her musical studio. In general, it is a novel field of research.

The focus of this thesis remains on assessing if action-driven and gesture-based interaction is considered by users as a *better* interaction model for sketching musical compositions than the *point-and-click* paradigm. *Better* is a subjective attribute, that can only be said through experience, and through interacting with the interface. Thus, it makes sense to conduct a usability study, task-based, comparing pairs of similar interfaces but with different interaction models, say, sketch-based versus traditional. Based on the traditional HCI of first identifying what are the test goals in usability testing (Nielsen 1993, Rubin 1994), in this case the aim is to evaluate the usability of the interfaces in terms of if they are easy to learn (*learnability*), easy to use (*usefulness*), satisfying to use (*experience*) and effective when used (*effectiveness*) (§4.2.3).

2.4 Tangible interaction

In this section we introduce the interaction model behind tangible user interfaces (TUIs), presented through some taxonomies developed since the 1990s. We also describe embodiment theories that surround these systems, where given the presence of physical objects on the interface invite users to be more bodily active. Finally, musical tables with tangibles are reviewed, whose interaction dialogue is close to the musical sketching nature.

2.4.1 Taxonomies

Since the early 1990s there has been an increasing amount of literature about physical interaction in the computer science field (Ullmer and Ishii 2000). Commonly known as *tangible interaction*, some researchers have provided theoretical taxonomies based on the experience gained developing these systems in order to grasp their behavior and improve their design (Klemmer 2004).

Ullmer and Ishii present the TUIs interaction model *Model-Control-Representation physical and digital* (MCRpd) (Ullmer and Ishii 2000). This model differs from the classical GUIs' one named *Model View Controller* (MVC) because while the latter separates the model (data) from the view (output) and the controller (input), the former employs physical artifacts to both control and represent digital information, coupling control and representation (see figure 2.3). In this model the MVC's view is separated into physical representations (*rep-p*) understood as the artifacts or tangible objects and digital representations (*rep-d*) which are media elements that support the visualization (e.g. video projection or audio).

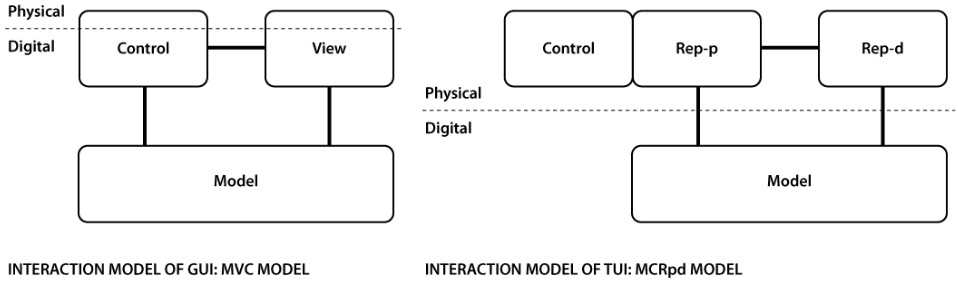

Figure 2.3: GUI interaction model and Ullmer and Ishii's TUI interaction model, based on figure from (Ullmer and Ishii 2000).

In addition, Ullmer and Ishii propose a conceptual framework for TUIs divided into four categories: *spatial*, *constructive*, *relational* and *associative*. Spatial systems highlight the position and orientation of the artifacts, constructive systems emphasize the modularity of building models, relational systems implies the interaction of almost two domains and associative systems consider weaker relations. Each TUI system analysed is tagged with its representation (*symbolic* or *iconic*), its functional roles (*containers*, *tokens* or *tools*) and if it supports *dynamic binding*. This pioneer and cited taxonomy provides an extensive overview of tangible systems, basically focused on the artifacts. According to this framework, the waveTable prototype (§3.1) is spatial because positions and orientations of the tangibles are relevant to the system (e.g. the eraser tool), relational because there are logical relations between tangibles (e.g. copy, paste or record tools) and associative for those relational tools that hold independence (e.g. open file); also it uses an iconic representation for the tangibles, named *phicons* (physical icons) when they imitate GUI icons.

Fishkin presents a taxonomy that relies on *metaphor* and *embodiment* (Fishkin 2004). First the term *tangibility* is defined as an attribute with multiple possible meanings. Then, the extent of tangibility is expressed by its relation with metaphor and embodiment parameters: the higher their values, the more tangible is going to be the system. Embodiment is explained as the integration of the I/O information, and in the framework there are four levels that range from *full* (or both channels completely engaged) to *distant* (or both clearly separated). For example, waveTable (§3.1) is based on direct manipulation, thus the feedback between input and output should be as tied as possible. Metaphor is highlighted by the author as a design element appropriate for TUI, proposing the following levels: *none* (or no metaphor), *noun or verb*, *noun and verb* and *full* (perception that the virtual system equals to the physical system). For example, waveTable seeks a full level of metaphor or transparent mapping: when the user erases a portion of the sound sample, that portion of the sound is erased. In sum, Fishkin's framework can serve particularly as TUI design guidelines.

Hornecker and Buur introduce a conceptual framework oriented to social and collaborative aspects of tangible interaction (Hornecker and Buur 2006). The framework is composed of four perspectives, each one contains a set of topics intended for providing organized knowledge based on their experience. The four categories are *tangible manipulation*, *spatial interaction*, *embodied facilitation* and *expressive representation*. Tangible manipulation is concerned with the artifacts and their degree of tangibility (*haptic direct manipulation*) or with the system feedback (*lightweight interaction*, *isomorph effects*). Spatial interaction arranges those concepts related to the environment, place, people and bodily actions (*inhabited space*, *configurable materials*, *non-fragmented visibility*, *full-body interaction* or *performative action*). Embodied facilitation categorizes those concepts that invite users to participate (*embodied constraints*, *multiple access points* or *tailored representation*) (§2.4.2). Finally, expressive representation deals with the representations of digital and physical elements common in tangible systems (*representational significance*) or with the user perception or decision making when interacting with these representations (*perceived coupling*, *externalization*). In this framework, the viewpoint is more centered on the users and the collaborative aspect than in the artifacts themselves. At times this categorisation is diffuse and general. This generality however allows greater flexibility and it can be either applied to multi-touch and tangible interaction.

Based on Hornecker and Buur's framework, using waveTable (§3.1) participants can move over the table the different objects (haptic direct manipulation), it can be done step by step with constant feedback (lightweight interaction) and the modification of both position and rotation of each tangible may have a visual and audio effect on the sound sample (isomorph effects). People standing around a table meet objects (inhabited space), there is a prime position that can see the waveform in the correct orientation of reading, albeit everybody can see what's happening (non-fragmented reality), and the actions imply body movement when drawing with a finger, copying and pasting or applying an effect (performative action). The physical set-up invite users to collaborate one after another (embodied constraints) and there are four access points shaped by the square form of the table (multiple access points). The amount of tangibles and the table set-up invite users to dialogue musically and verbally using the objects (externalization), and a link between physical and digital representations is established through real-time visual feedback (perceived coupling).

2.4.2 Embodied interaction

Bodily actions and cognition result in embodiment, which means not only the degree that user thinks the system is integrated with the object manipulated, but also the social and physical environment where the interaction takes part (Fishkin 2004), usually related to tangible interaction since the physical and the digital worlds are connected by learning through user actions.

With desktop computers users perform a varied range of activities like writing and essay, playing a song or chatting with a friend maintaining the body in the same posture (Klemmer et al. 2006). According to the authors, this scope contrasts with how the body differs when riding a bicycle or playing drumms, for instance. Theories from psychology, sociology and philosophy encompass a connection between perception, cognition and action: The "Thinking through doing" quotation stresses the relationship between bodily activities and cognition, known as *embodied interaction*. This approach integrates gestures, physical objects and actions: the user manipulates artifacts to accomplish tasks (*pragmatic action*) or to explore the context of the tasks (*epistemic action*) resembling infancy periods of learning. The authors illustrate embodied interaction with the example of thinking through prototyping, where product design is done by working iteratively on a prototype, immersed in a continuous creative process based a collection of dialogues with materials: sketching on paper or shaping clay, for instance.

Considering the influence of embodied interaction, some design guidelines are proposed that emphasize contexts, subjects and embodied actions. According to Fernaeus et al., the tangible interaction practice has implied a move away to a new set of ideals with design consequences in tangible interaction as well as HCI: *from* *information-centric to action-centric*, *from properties-of-system to interaction-in-context*, *from individual to shareable*, *from objective to subjective interpretation* (Fernaeus et al. 2008). The foundations of embodied interaction and a set of design guidelines based on tangible and social computing are proposed by Dourish, being a reference when facing embodied interactive environments (Dourish 2001).

2.4.3 Musical tables with tangibles

A number of systems that exploit tangible interaction are designed for musical composition and live music performance. Most of them consist in musical tables with tangibles, a setting that promotes collaboration and creativity, among others (Jordà et al. 2007). The Physical sequencer, one of the instruments of Enrico Costanza's Audio d-touch (Costanza et al. 2003) interface, represents sound samples as tangible objects in a cyclic timeline. These sounds can be loaded into objects using a microphone and several audio effects can be applied in real time. The Music Table (Berry et al. 2003) is a sequencer that allows to compose musical patterns by positioning cards on a table. Cards are tracked with a top-camera and displayed on a separated screen adding an augmented reality layer. There is a copy card that enables copying patterns to phrase cards that can be further reused or edited without needing the original note cards. The reacTable (Jordà et al. 2005, Kaltenbrunner et al. 2006, Jordà et al. 2007) has become one of the most popular multi-touch and tangible tabletop instruments. This collaborative and rounded instrument incorporates dynamic patching in a similar way than traditional modular synthesizers. Among many other features, the reacTable enables to draw the waveform using one finger and looping samples is also supported by means of sampler objects (Jordà et al. 2007). Using the reacTable technology, the scoreTable (Jordà and Alonso 2006) is a music score editor that explores real-time symbolic composition in a circular stove. The system is more focused on musical composition allowing users to move notes on the stove. Golan Levin's Scrapple (Levin 2006) system enables to create an spectrographic score using tangible objects laid on a long table, with an augmented reality layer for visual feedback. A compromise between compositional precision and improvisation flexibility is sought in the tradition of spectrographic musical composition.

While these systems are focused on sound generation or real-time sequencing, they address tangentially the issue of tangible sound editing, an activity suitable for tangible interaction: it is a daily and organized task that can become highly creative when exploring alternative techniques. Table 2.1 exposes a relation between the tangible systems reviewed and the principal features of a desktop sound editing tool.

Sound editing	Draw	Apply Effects	Record Input (Sampling)	Record Output	Copy-Paste	Loop	Zoom
(Costanza et al. 2003)		•	•			•	
(Berry et al. 2003)		•		•	•	•	
(Jordà et al. 2007)	•	•				•	
(Jordà and Alonso 2006)		•			•	•	
(Levin 2006)						•	

Table 2.1: Sound editing operations used in cited references.

The waveTable prototype (Roma and Xambó 2008) described in this thesis (§3.1) proposes to use the tangibles as tools representing functions that operate on data instead of using the tangible objects as representations of data as these systems do. This approach may facilitate the implementation of basic editing operations often found in desktop computers resulting in a real-time tabletop sound editor.

There are some design issues highlighted by the reviewed literature worth to be summarized.

- **Visual feedback**. There is a common insight in making interactions legible for the observers (Patten et al. 2002) although the strategies followed can range from integrating a visual layer on the table highlighting each object using a video projector underneath (Jordà et al. 2007, Levin 2006, Jordà and Alonso 2006) or on top of the table (Levin 2006) to separate the visual layer on a frontal screen (Berry et al. 2003), as well as exploiting directly the physical objects getting rid of any additional visual layer (Costanza et al. 2003).

- **Compositional precision**. It is addressed, apart from the visual feedback employed (Jordà et al. 2007, Levin 2006), using certain tools that will depend on each musical instrument. In the case of the sequencers, there are common tools employed such as a current time indicator (Levin 2006) or a grid tool to allow more time and pitch precision (Levin 2006, Costanza et al. 2003).

- **Improvisation flexibility**. The will that everything is feasible, immediate and intuitive combined with the flexibility of the sound design algorithms (Jordà et al. 2007) or preserving the possibility for body-based improvisation (Levin 2006) are some of the strategies undertaken.

- **Collaboration**. This aspect is partly solved using large squared surfaces (Levin 2006) or rounded ones which imply no leading voice (Jordà et al. 2007, Jordà and Alonso 2006). Those small and squared tables are limited to a solo performance (Costanza et al. 2003, Berry et al. 2003).

2.5 Pen-driven interaction

In this section we introduce pen-tablet mappings from gestures to sounds and sketches, first observing how different pen-tablet approaches define an alphabet of gestures with sonic correspondences, and second analysing how related is the interaction of pen-driven sketching systems at the early stages of design to the early stages of the music composing process. Then, a review of some electronic music scores gestural-based is presented.

2.5.1 Alphabet of gestures

Studies of the correlations between gestures and drawings using pen-driven or stylus-based input devices are common. There is a general will to establish an alphabet or collection of atoms that enables to build words and sentences. One approach inspired by Paul Klee's Pedagogical Sketchbook (Klee 1968) is based on applying the visual granularity of the trivium point-line-plane to elementary gestures and sounds: for instance, straight gestures produce linear sounds while circular gestures produce curvilinear sounds (Goina and Polotti 2008). Golan Levin explores audiovisual real-time performance by developing a set of painterly interfaces based on the similarity between pen-driven interaction and the artistic activities of painting or drawing, allowing users to maintain an expressive gestural dialogue (Levin 2000). Another approach is based on the user practice on stylus dexterity and expressivity (Zbyszyński 2008).

In a three-dimensional space, the traditional six *degrees of freedom* (DOF) or possible movements and orientations are the three lateral movements along the *x*-, *y*- and *z*- axes respectively, and the three rotational movements around these same axes *rx*, *ry* and *rz* (Bongers 2006). Bert Bongers suggests that an input device with more DOF is better suited for accomplishing complex tasks than another with less DOF.

Concretely, when comparing DOF of a mouse and a pen-tablet, the author describes that while the mouse has two DOF in the x- and y-axes, one rotational DOF for scroll wheel and some discrete switches, the tablet is able to identify the pen position in the x- and y-axes, tip pressure and the angle with the surface. According to the author, the point-and-click paradigm imposed by the mouse is time-consuming for performing complex tasks, rather the pen is more precise and adequate for these tasks. Bongers has developed the *Physical Interface Design Space* (PIDS) framework to analyse physical interfaces (§3.2.2).

Pen-based interaction fits well with the design process of *computer-aided design systems* (CAD), specially in sketching activities carried out at the early stages of design (Dickinson et al. 2005). According to Dickinson et al., the current CAD paradigm is principally based on the mouse and the WIMP interface. Although the 3d mouse has achieved 6 DOF around the x-, y- and z-axes that are mapped to zoom, pan and rotation facilitating the control to navigation and position, those constraints of mouse-driven interaction remain: supporting principally the creation of precise and detailed models (*structured interaction*) or producing RSI injuries. According to the authors, pen-driven tablet is able to track not only the position of the stroke but also the details of the stroke (pressure, the angle of the pen, the speed, which end of the pen is being used and, if the pen has buttons, which buttons are being pressed during the stroke). That precision can contribute into the creative stages of design given its similarity to the pen and paper interface already used in this creative stage (*creative interaction*).

The author illustrates a spectrum of pen-based interactive paradigms in the literature (see figure 2.4) for creating interpretable models, ranged from creative to structured interaction. *Gesture based* is the closer to creative interaction, because the user can draw virtually any shape although there is a finite set of gestures recognisable. *Visual notation recognition* assumes that a diagram and interpretable elements are sketched on the tablet. *Structured gesture* is the closer to structured interaction before *WIMP-based*, and while the former uses gesture-based control for executing operations, the latter employs dialogue boxes and menus for that purpose. After analysing these interaction paradigms applied to CAD design-related tasks, the author concludes that current pen-tablet paradigms still don't offer complete solutions for model creation, although it deserves further investigation given its potential in the creative stages of design.

Hence, we observe commonalities between the refinement process of music composing and designing, therefore the spectrum of input paradigms can put into con-

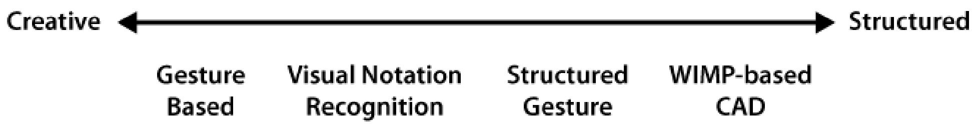

Figure 2.4: Spectrum of pen-tablet input paradigms, based on figure from (Dickinson et al. 2005).

text pen-driven systems of other creative domains. For example, the wiiteBoard prototype (§3.2) addresses the brush-canvas-like feel allowing to freely draw by gestures any shape on the digital canvas using a light pen, interaction paradigm that in the spectrum would be situated nearby creative interaction.

2.5.2 Electronic music scores gestural-based

The idea of hand-drawn sounds is already envisioned in the early 1920s by László Moholy-Nagy, who proposed the transformation of the phonograph from an instrument of musical reproduction to an instrument of musical production, (available in (Moholy-Nagy 2004)):

> ′′ An extension of this apparatus for productive purposes could be achieved as follows: the grooves are incised by human agency into the wax plate, without any external mechanical means, which then produce sound effects which would signify without new instruments and without an orchestra - a fundamental innovation in sound production (of new, hitherto unknown sounds and tonal relations) both in composition and in musical performance.´´

The author proposes a set of experiments with the phonograph for musical composition based on finding methods for working on large-scale groove-script records by hand that can be reduced and played later on by the phonograph. In the 1970s, some systems for musical composition allowed to draw waveforms and envelopes such as Iannis Xenakis' UPIC or the Fairlight CMI[9] using pen-driven interaction. UPIC explored different techniques for sound generation from drawings (Marino et al. 1993). The system allowed users to draw lines, curves and points on a time-frequency score using a graphic tablet for input.

[9] http://en.wikipedia.org/wiki/Fairlight_CMI

Currently there are systems such as Hyperscore, a graphical computer-assisted composition system devoted to users musically untrained (Farbood et al. 2004). Users can create musical compositions on a canvas using a windowing system, thus remaining in the UI design of the desktop metaphor (§2.1.3). The window's vertical axis represents pitch while the horizontal one represents time. The notes are represented by droplets which are drawn by the user clicking on the grid, and which may be modified later on. In addition, it is possible to add pitch envelopes drawing curves and bends in a line. Each droplet colour represents one voice with its own window, and there is also a complete window view with the sum of all voices. Hyperscore offers a set of traditional chords to be applied that leads to learning classical music, but with little room for other music genres.

Notice that the majority of these audiovisual systems rely on the two-dimensional spatial-temporal mapping where y-axis corresponds to frequency (pitch) and x-axis corresponds to time (duration). Within this tradition has been built the wiiteBoard prototype (§3.2).

2.6 Mobile interaction

In this section we describe some projects that explore multimedia mobile interaction and sketching. On one hand, we introduce the concept of mobile multimedia ideation in the form of diary field studies. On the other, we outline the notion of mobile music ideation reviewing some innovative portable computer music instruments.

2.6.1 Multimedia field studies under mobile conditions

Mobile computing devices are everyday more powerful in terms of processor performance, storage capacities and flexibility of working wherever with easily access to internet (Blackwell et al. 2007, Boll 2005). They can be considered either a social communication tool or a daily working tool. Multimedia mobile applications are becoming more popular for users, and the challenge now is to achieve real user-centered mobile multimedia applications (Boll 2005).

Some experiments employ these devices as portable field work tools. A technique for capturing information under mobile conditions is explained by Brandt et al., who propose a technique for multimedia ideation, a diary study based on users sending to a server small snippets of text, audio or pictures (Brandt et al. 2007). Later on, the participants can access to a website and continue in the form of diary entries. The novelty of this study is the technique proposed about performing diary studies under mobile conditions during a long-term, that can provide more consistent data when compared to the classical usability lab. In addition, the open structure of the experiment is closer to the real world. Another advantage is that almost all users have a mobile and they know how to use it. The authors assess the system with participants, and the results show that overall media usage is inclined towards text rather than audio or pictures since it is the most confident medium for them.

2.6.2 Portable computer music instruments

Usually technological limits force the software designer to use alternative creative solutions. Music software in handheld and portable devices is constrained by these limitations, emerging alternative interfaces that invite to produce music by experimentation. These applications are more focused on the fast production of sound content, usually having text-free interfaces, supporting real-time performance and providing visual feedback. That is the case of nanoloop[10], designed to run in old Game Boy consoles and which is stored on a cartridge like a game (see figure 2.5). The interface is based on a 4*4 matrix, which corresponds to 16 steps and where events are represented graphically allowing the creation of patterns. The current step is highlighted in dark grey, and it moves through the matrix continuously. The notes can be set or removed in real-time pressing one button and modifications on them (volume, pitch and so forth) are displayed graphically, as well.

The "Pure Data anywhere" (PDa) project aims at using mobile devices as computer music instruments (Geiger 2003, Geiger 2006). The virtual guitar and The virtual drum set are two computer music instruments developed by Geiger for PDA's using the PDa language that explore the affordances and possible mappings of these devices using the touch screen as a sophisticated controller. The four interaction principles followed are: *region based triggering*, *gesture recognition*, *border crossing* and *continuous parameter control*. However, patching on the handheld device itself, although possible, it is complicated (Geiger 2003) (§3.3.2). With the advent of popular multimedia devices such as the portable media player iPod[11] or the multimedia phone with a multi-touch screen iPhone[12], the development of creative computer

[10] http://www.nanoloop.com
[11] http://en.wikipedia.org/wiki/IPod
[12] http://en.wikipedia.org/wiki/IPhone

Figure 2.5: Nanoloop running in the old Game Boy pocket console.

music software is increasing (e.g. RjDj[13], PdPod[14], BeatMaker[15] or iDrum[16]).

Finally, we find as well standalone portable instruments that take full advantage of mobility. TENORI-ON (Nishibori and Iwai 2006), for instance, is a standalone portable electronic instrument, that operates with a graphical matrix interface for creating visible music composed of a 16x16 light-matrix of LED switches. Each switch carries a sound that can be loaded and played in loop mode. The system can synchronize with another one being possible to make music collectively in real time.

[13] http://www.rjdj.me
[14] http://www.ipodlinux.org/PdPod
[15] http://intua.net/products.html
[16] http://www.izotope.com/products/audio/idrum

Chapter 3

Prototypes

In this chapter we present three prototypes which are all based on traditional electronic music tools, say, an editor, a sequencer and a sampler, specially conceived for the early stages of the creative process. A description of each system is done in terms of concept, implementation and open design issues.

3.1 The waveTable editor

3.1.1 Concept

The author and Gerard Roma have developed an audio waveform editor that can be operated in real time through a tabletop interface (Roma and Xambó 2008). It consists in a prototype started as a classroom project developed in the courses of *Interactive Systems Workshop* by professors Sergi Jordà and Martin Kaltenbrunner and *Electronic Music Workshop* by professors Sergi Jordà and Günter Geiger at the Computer Science Faculty of the Universitat Pompeu Fabra during 2006-07. The system combines multi-touch and tangible interaction techniques in order to implement the metaphor of a toolkit that allows direct manipulation of sound. In our system tangibles are used as tools, which represent functions that operate on data physically applied to the projected sound wave. The nuances able to be performed on an evolving loop drive the instrument to be well suited for live performance and sound design, the latter specially in the early stages of composition (see figure 3.1).

3.1.2 Implementation

waveTable is based on a reactable-class hardware that consists in a table with a translucent surface that has a camera and a projector beneath. While the former is continuously tracking the position and orientation of tools and fingers situated

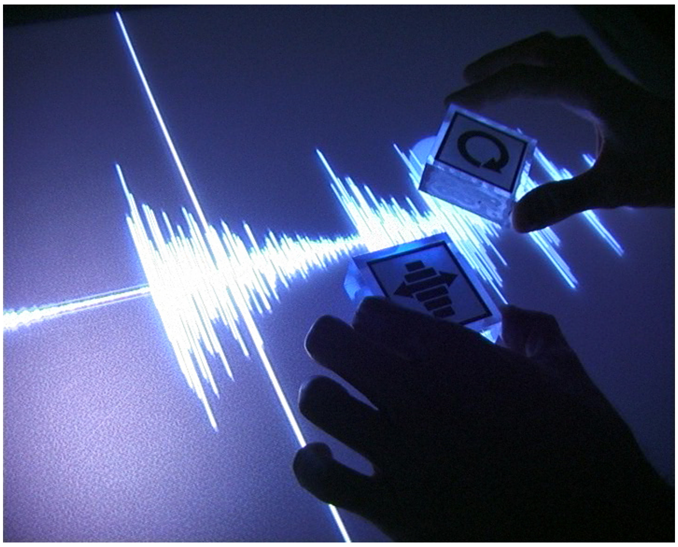

Figure 3.1: The waveTable editor prototype.

on top of the surface using the reacTIVision framework, the latter displays the visuals in synch with sound using the SuperCollider language that can provide real time audiovisual feedback. The UI design is inspired by the toolkit metaphor of desktop applications, being detected the 2D position, rotation and presence of each tool, providing visual feedback as well (see figure 3.2).

Hardware

The hardware has been built upon the premise of using low-cost components and following the discussions in the reacTIVision forum[1], the tutorial[2] and news posted both by Harry van der Veen[3] and the Natural User Interface (NUI) group[4], and of course the reacTable publications (Jordà et al. 2005, Jordà et al. 2007) and its website[5].

- **Table and Surface**. The surface ought to fulfill a compromise between being transparent enough allowing the camera to detect the fiducials and opaque

[1] http://sourceforge.net/projects/reactivision
[2] http://www.multitouch.nl/documents/multitouchdisplay_howto_070523_v02.pdf
[3] http://www.multitouch.nl
[4] http://nuigroup.com
[5] http://reactable.iua.upf.edu

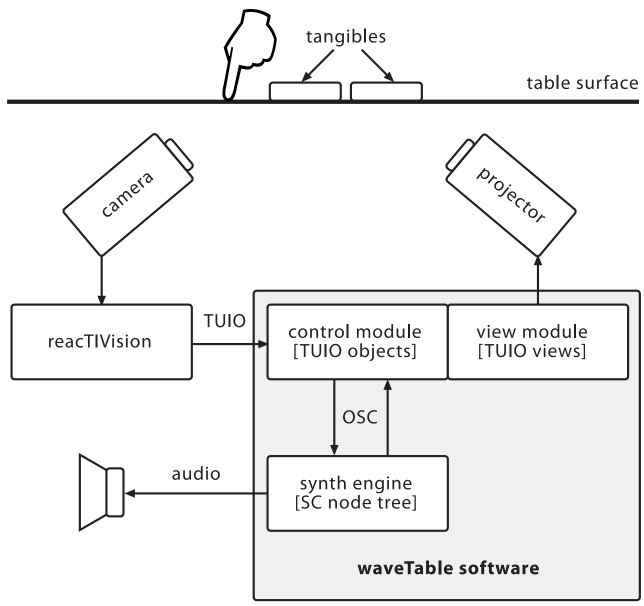

Figure 3.2: An overview of the waveTable system.

enough allowing the projector to display the projected images (see figure 3.3). In addition, the material has to be sufficiently heavy to support fingers and fiducials but also light enough to allow portability. Our option is based on a translucent surface of plastic with approximately 2 mm of width. The dimensions are 50 by 50 cm.
We decided to build an iron frame with easily dismountable pieces in the junctures that allow us mobility (see figure 3.4). There are a total amount of eight bars whose dimensions are: 4 bars of 1 m and 4 bars of 50 cm, resulting in a 1 by .5 by .5 metres frame. The translucent surface is aligned with magnets, and the laterals are covered with opaque textile (traditionally used in homemade photo lab).

- **Camera and Lighting**. The webcam Phillips SPC 900 has been modified to a webcam that is able to detect infrared (IR) light. This can be done by replacing the IR-block filter (common in these webcams which enables to detect all visible light except the IR) by an IR-pass filter. Due to the difficulty in this camera

Figure 3.3: The waveTable structure: Iron frame.

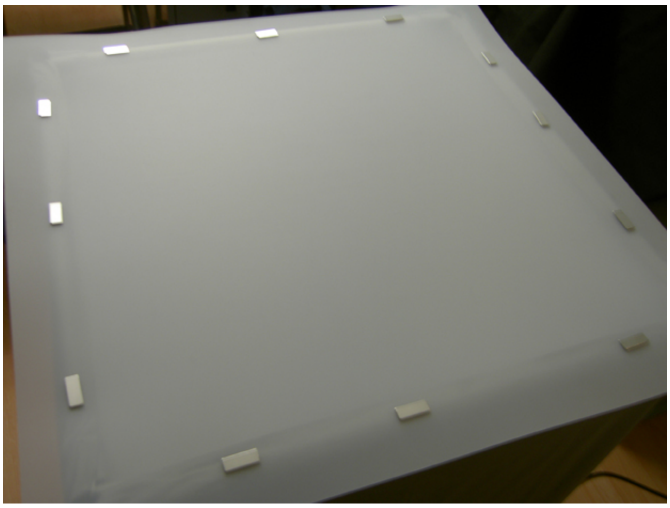

Figure 3.4: The waveTable surface: Plastic with magnets on iron frame.

to remove the single IR-block filter because it is attached to the lens, we have replaced the original lens with another one without IR-block filter, a CCTV

lens IR 4.3mm. Finally, a couple of black photo negatives acting as IR-pass filter are attached in front of the lens in order to only see IR and remove all the visible light. This step is necessary to allow the detection of fiducials and fingers using reacTIVision.
For the lighting layer, 120 IR LEDs Infrared Emitter 5mm (Compl. BPW40-SFH300) have been solded on 3 boards powered by a 12 volt adapter plus using four directional IR focus Cebek C-2290 52mm. The considerable height of the table as well as working with standard LEDs are the main reasons for working with such amount of LEDs (see figure 3.5).

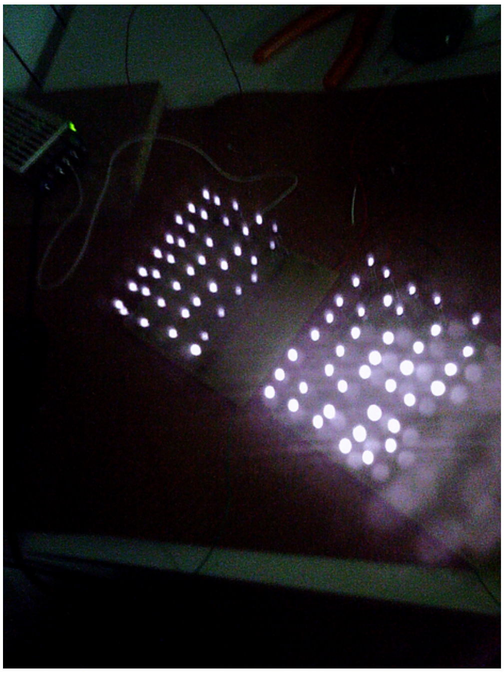

Figure 3.5: The waveTable lighting: 80 IR Leds (2 boards).

- **Projection**. The projector is employed to display the image from the computer to the table surface. The projector has a 4:3 ratio and is situated underneath the table, allowing direct manipulation of the image. We use a mirror to reflect the image that comes from the projector to the surface with the aim to occupy as much as possible the same dimensions of the table (see figure 3.6).

Figure 3.6: The waveTable projection.

Software

Tools are acrylic plastic pieces that have attached on one side fiducial markers. Tools and fingers are captured by the webcam and processed with reacTIVision, which tracks position and rotation of the fiducial markers as well as position of the fingers. This data is encoded using the Tangible User Interface Objects (TUIO) protocol based on Open Sound Control (OSC) (Kaltenbrunner and Bencina 2007) and sent later on to the waveTable software.

The software is developed using SuperCollider (McCartney 2002), a computer music language and free software based on a client-server model. There is a server, the synthesis engine *scsynth* managing the audio synthesis, and a client, the language engine *sclang*, and they communicate via OSC. This environment allows to operate on audio buffers in real time and in the Mac OS X version it is possible to draw graphics primitives and waveforms. External control in the server is possible via OSC, which allows an integration with reacTIVision using Till Bovermann's implementation of TUIO protocol[6].

The software is divided into control, model and view modules. The control module is composed by a hierarchy of TUIO objects that handle each of the tools and a

[6] http://tuio.lfsaw.de

single class that handles the TUIO cursors (fingers). The model is implemented as a SuperCollider (SC) server node tree that runs synth definitions and manages to play the sound buffer and the dynamic application of effects. Lastly, the view module is another hierarchy of objects that manages the graphic representation of tools, envelopes and the waveform display.

Interaction Design

The UI design is based on the toolkit metaphor commonly found in the graphical desktop applications, namely tools palette (see figure 3.7). In waveTable each physical object represents one tool. The computer vision detects 2D position, rotation and presence of the tools as well as the motion of one or two fingers. Each tool is activated by a concrete gesture related to the tool's meaning: i.e. the *Eraser* tool deletes part of the sample when moving it along the x-axis or the *Two finger zoom* allows navigation between the closest and the most general view of the sound sample depending on if they are moved to an inner or outer direction along the x-axis (Roma and Xambó 2008). There are editing, effects, and file tools, as well as visualization and navigation gestures and tools. The editing tools represent operations of basic modification on the sound (erase, draw, copy, paste, gain). The *Effects* tools correspond to common audio effects applied to the sound in real time (delay, low pass filter, tremolo, reverb and bit reducing). The *File* tools represent operations of managing the sound file (open file, play and record). Finally, the visualization and navigation gestures and tools are related to displacement and zoom level (two finger zoom, one finger scroll and grid).

Visual feedback is provided through a glowing halo around each tool, that indicates that the object has been detected by the system. In addition, there is a *current time indicator* that indicates the exact moment of the sound being played. The *Open file* tool displays an endless radial menu allowing to preview samples from a collection, pointing one with a finger enables to load it. In the case of situating on the table the *Record* tool, it shows three states: an initial state with a grey halo around indicating that it is waiting to record at the next starting loop, a second state that together with the *current time indicator* get red meaning that it is capturing the output of the system in real time, and a third and last state that all the effect tools together with the record tool are deactivated, indicating that the playback sample has been swapped for the result and all the objects can be removed from the table. In the case of the effects, each one displays one envelope of its most important parameter, represented graphically with a particular colour. Envelopes are composed of two sinusoidal segments displaying a smooth curve, that is modified depending on the position and rotation of the object according to the zoom level.

In sum, the selected sample is played in a loop which turns sound editing into a composition process that together with the visual feedback are well suited for live performance.

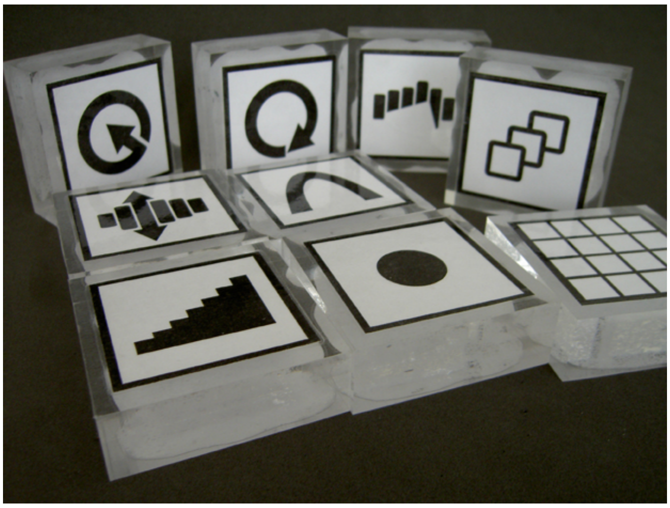

Figure 3.7: A set of the waveTable's tangibles.

3.1.3 Issues

Informal observation has provided us a set of open UI design issues worth to be commented:

- **Portability**. Given that it is an ideal instrument for live performance, the current size of the prototype should be reduced and the components should be compacted.
- **Mapping**. Currently each physical object or tool corresponds to an operation, which is identified by an icon text-free. The question that arises is if the user will be able to identify all the tools if the collection grows.

- **Functionalities**. Some tools need to be added such as a *Snapshot* tool to save the sound buffer to the hard disk or a *Crop* tool to isolate specific segments of larger sound recordings.

3.2 The wiiteBoard sequencer

3.2.1 Concept

The wiiteBoard sequencer is a spectrographic sequencer with gestural-driven interface that allows to paint on a screen using a custom light pen in a similar way than brush-on-canvas or light-on-blackboard. As a result it is possible to sketch an score composed of sine wave frequency paths and to make electronic music from scratch, in real time. This prototype has been developed by the author and Gerard Roma in the course *Advanced interaction design* by professor Bert Bongers at the Computer Science Faculty of the Universitat Pompeu Fabra during 2007-08 (see figure 3.8).

Since the early electronic music age, an alternative representation of music was sought due to traditional scores could not represent properly the possibilities of those novel electronic devices. Electronic music pioneers such as Stockhausen or Xenakis (*Metastasis*, 1954 (Xenakis 1992)) sought alternative representations of music exploring the possibilities of the electronic music scores. Spectrograms or diagrams of the frequency content of sound, display how sounds can be represented as a sum of sinusoids. This visualization has often been used for graphical interaction with sound and music in real time: from commercial applications such as MetaSynth[7] or Spear[8] software to artistic projects such as Golan Levin's Scrapple (Levin 2006), a tangible spectrographic score table (§2.4.3). These spectrographic scores provide a two-dimensional representation where the y-axis represents the frequency and the x-axis represents the time. Thus, with an spectrographic interface it is able to compose and create sound starting from a blank canvas, which has been a repeated concept in computer music (Levin 2006), in the tradition of Xenakis' UPIC system (Marino et al. 1993).

[7] http://www.metasynth.com
[8] http://www.klingbeil.com/spear

Figure 3.8: The wiiteBoard sequencer prototype.

3.2.2 Implementation

Users can draw notes (strokes) on a score (computer screen) in real time while the melody is played in synch. The hardware consists in a computer screen, an IR light pen and the Wiimote camera, while the software is based on two modules, the *graphics* module that manages the strokes visualization and the *sound* module that manages the sonification. Then the physical interface design of the light pen is explained using the PIDS taxonomy.

Hardware

For this section, we have followed Johnny Lee instructions about building low-cost multi-point interactive whiteboards[9]. The hardware of the system requires three elements: an electronic blackboard, the Wiimote IR camera and an IR light pen to

[9] http://www.cs.cmu.edu/˜johnny/projects/wii

draw. For the first component, a laptop screen works perfectly well. Additionally, it can be overlaid a transparent plastic screen-size in order to avoid damaging the screen. The Nintendo Wii Remote camera is situated on top of the screen. After connecting Wiimote to the laptop via Bluetooth, the Wiimote Whiteboard software[10] (Mac version) allows to calibrate and track the IR light pen, a low-cost one that we have built for that purpose (see figure 3.9).

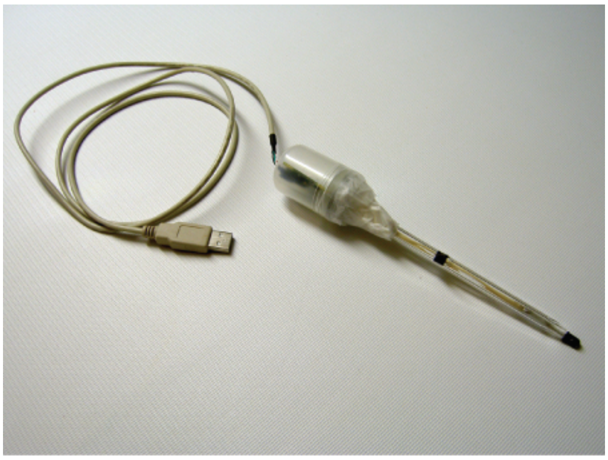

Figure 3.9: The wiiteBoard sequencer's lightpen.

Software

The wiiteBoard software contains two modules: the *graphics* module and the *sound* module. The graphic module has been developed using Processing (Reas et al. 2007), an open source programming language, and it manages the visualization that comprises the strokes drawn but also the current time indicator. Meanwhile, the sound module which has been developed using SuperCollider (McCartney 2002), and that is connected to the graphic module via OSC using the oscP5 library[11] manages the sonification of the strokes drawn depending on the frequency.

[10] http://www.uweschmidt.org/wiimote-whiteboard
[11] http://www.sojamo.de/libraries/oscP5

Physical interface design

Bert Bongers proposes a taxonomy for physical interface analysis, namely the Physical Interface Design Space (PIDS) (Bongers 2006). It is well suited for physical interfaces that employ movements in the input and output. In our case, we will concentrate on the input. There are three dimensions to be considered: *Range*, *Precision* and *Haptic Feedback*. *Range* concerns with the space of influence from a human scale: *within the hand*, *within the reach of the arm* or *within the architectural space*. In the wiiteBoard prototype the use of the pen to draw on the screen results in being within the reach of the hand, it pertains therefore to a body-sphere range. *Precision* is related to resolution (in bits), sample rate and latency. In our prototype the resolution is given by the Wiimote camera which provides a resolution of 1024x768 pt. The third parameter, *Haptic Feedback*, is concerned with how it feels when touched and the input effort required. In that case, wiiteBoard does not have haptic feedback, only a button as input controller.

The wiiteBoard's light pen has two DOF: the x-axis mapped to time, and the y-axis mapped to frequency. We have done informal tests adding an accelerometer sensor[12] to the light pen summing two more DOF: the *rx* mapped to the colour opacity that modifies the amplitude and the *rz* mapped to the stroke width that modifies the timbre. From an ergonomic viewpoint, it is uneasy to rotate the light pen on the x-axis while drawing, however the rotation on the z-axis is less uncomfortable. Nonetheless we have removed this sensor being pendent further tests in this direction.

3.2.3 Issues

Informal observation has provided us a set of open UI design issues worth to be commented:

- **Performance**. Currently it is difficult to have a complete control of the light pen and it is only possible to draw from left to right. These aspects should be improved.

- **Tilt**. Preliminary tests using an accelerometer sensor on the light pen points out that can be interesting to investigate possible correlations from tilting strokes to sound.

[12] http://www.phidgets.com

- **Logarithmic scale**. An application of the logarithmic scale in the frequency representation would improve the precision at the perceptual scale of the range of notes.

3.3 The tr4ck recorder

3.3.1 Concept

tr4ck is a mobile multitrack recorder for recording and mixing up to four tracks simultaneously using a handheld device. The system allows to record from scratch with the built-in micro up to four sounds, replay and blend them in realtime (see figure 3.10).

Figure 3.10: Close view of musician playing the tr4ck prototype with Nokia N810.

The early four track recorders based on an audio-cassette tape (e.g. the TASCAM Portastudio[13]), were creative tools used mainly during the 1980s and 1990s for sketching musical compositions in the form of demos and lo-fi recordings, and sometimes even commercial albums. It allowed to record sounds on each of the tracks and mix them.

[13] http://en.wikipedia.org/wiki/Portastudio

With the advent of digital music since the 1990s, there were some handheld music workstations such as the Yamaha QY10[14], which was focused on enabling musicians to compose music while travelling. It had a MIDI sequencer but recording was not possible. On the other hand, there were phrase sampling instruments, both stand-alone (e.g. Yamaha SU10) but also as a module within a bigger instrument such as a groovebox[15], that could offer a 4-track audio looper with mixer (e.g. Roland MC-303 or MC-09), more conceived for performance (but also used in the studio), than for the mobile studio (see figure 3.11).

Figure 3.11: Roland MC-09 Phraselab.

Solid state cards, however, have allowed since the 2000s the advent of handheld digital audio recorders (such as Edirol R-1 or Zoom H2[16]). These portable recorders allow musicians to record field recordings, interviews, demos or podcasting among others, but in case they support multitrack recording, it is difficult to mix them in an immediate way (see figure 3.12).

tr4ck takes advantage of both, the handheld recorders more focused on composing and the portable instruments more focused on live performance. The system allows the user to sketch audio loops while moving around or travelling.

[14] http://en.wikipedia.org/wiki/Yamaha_QY10
[15] http://en.wikipedia.org/wiki/Groovebox
[16] http://en.wikipedia.org/wiki/Zoom_H2

Figure 3.12: Zoom H2 digital audio recorder.

Sampling sounds is directly associated with Pierre Schaeffer's *musique concrète* around 1950s, who composed music by creating montage from the accidentally found and stored *objet trouvé* (de la Motte-Habber 2002), being considered the godfather of sampling composition (Cox and Warner 2004). This approach implies that everyone can create music reordering the sampled sounds found in an everyday environment using the two concepts of *cut* and *mix*. Thus, to record is to cut a sample from its original context, and to mix is to provide the sample with a new meaning (Cox and Warner 2004). Since the end of the 1960s the term of *soundscape* (Schafer 1993) has been used in reference to the sound of the city as a social organism where there is a specific relationship between people and their urban space, and a deeper understanding of the culture and society can be done by examining their sounds (Barthelmes 2002). The tr4ck prototype enables the musician to explore internal and external soundscapes through *soundwalks* cutting and mixing samples in a similar way than writing down ideas in a small notebook.

3.3.2 Implementation

tr4ck is a computer music patch written in PDa that consists in four tracks able to record and play simultaneously. The UI design is constrained to the PDA screen resolution and the stylus-based interaction, using a minimal aesthetics of sliders and buttons.

Software

The tr4ck recorder consists in a patch written in PDa, a core version of the computer music system PD (Pure Data) that runs on PDA's with Linux support (Geiger 2003). PDa, like PD, is a real-time computer music language and it is free software. tr4ck has been developed on the Linux based internet tablet Nokia 810 using the precompiled package of PDa[17] for Maemo platform[18].

There are four *tracks*, each of them has one *recorder* module and one *player* module (see figure 3.13). While the recorder module manages the operations of recording, the player module is focused on the reproduction functionalities of the track. Given that PDa does not support input, the recorder module uses the library *gst-streamer*[19] for that purpose. The player module employs the builtin *sfread~* object which enables to manage the reproduction of sound samples.

UI design

The UI design is constrained to the PDA screen resolution of 800x480 px. The interface displays a matrix of two by two blocks or tracks, each of them with three large buttons that respectively record, stop and play a sound sample, one horizontal slider that manages the sound volume and another vertical one that defines the loop length (see figure 3.14). The usage of the program is simple: The user carrying the PDA clicks on the record button with the stylus when she wants to record, then clicks on the stop button to end the recording and next clicks on the play button in order to listen to the recorded sound. This action can be repeated for the rest of the three tracks, and if the user wants to record again one of the previous tracks, the procedure is exactly the same. The volume of each track can be modified and it is possible to activate a different loop length than the own sound sample length. The resulting sum of sounds is being played in a loop.

[17] http://gige.xdv.org/pda/release/maemo
[18] http://www.maemo.org
[19] http://www.gstreamer.net

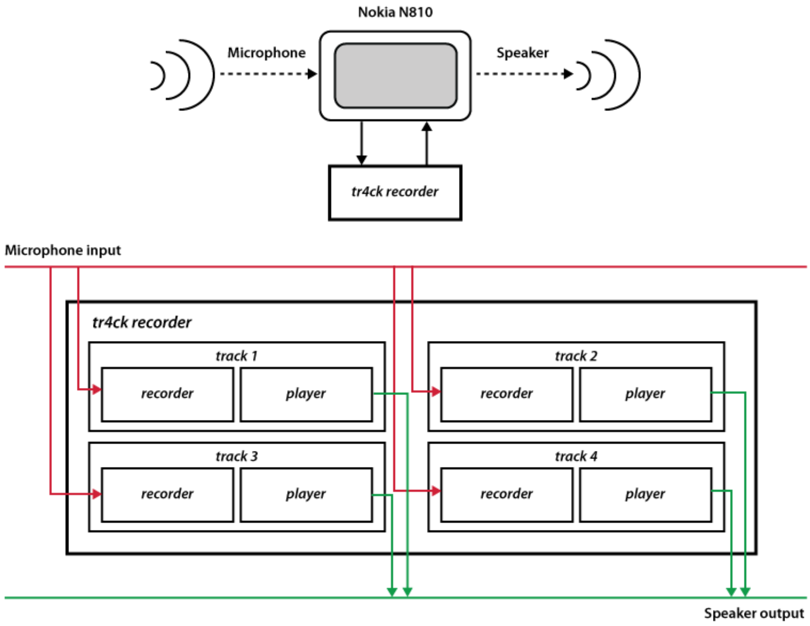

Figure 3.13: The tr4ck recorder components diagram.

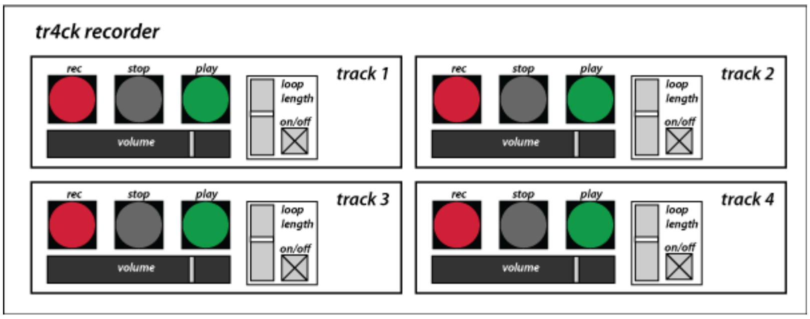

Figure 3.14: The tr4ck recorder UI design.

3.3.3 Issues

Informal observation has provided us a set of open UI design issues worth to be commented:

- **Performance**. Some drawbacks should be fixed: less required period between stopping the recording and playing it, or improving the current recording sound quality if possible (currently 16bits-8000Hz).

- **Functionalities**. The current version only incorporates the most basic functionalities of a sampler: *record*, *stop*, *play*, *volume* and *loop length*. In our opinion, additional editing features such as being able to modify the velocity of each track or to store the file in a single sound sample are the most prominent ones to be implemented.

Chapter 4

Evaluation

In this chapter we begin by introducing the usability testing practice. Then, we continue with a detailed explanation of the usability study to be conducted in terms of main objective, methodology to be followed according to standard measurements and a brief description of the testing set-up. We end with an overview of the interfaces to be compared and the tasks to be performed in the usability test.

4.1 Usability testing

Usability testing is a common practice in user-centered design that involves users in different stages of the design process. While users perform representative tasks, they are observed using different techniques that facilitate gathering empirical data. The techniques used (task-based user testing, think-aloud, questionnaires or interviews, among others), are a common practice taught in standard texts such as (Nielsen 1993, Rubin 1994, Cooper et al. 2007b).

4.2 Usability study

4.2.1 Objective

The goal of conducting a usability study is to assess that sketch-based systems for composing music are adequate tools to enhance musical creativity, both for expert and novel musicians. This approach includes TUIs, pen-driven or stylus-based interfaces including those designed for mobile devices.

4.2.2 Methodology

In order to assess these systems, a comparative usability study is conducted. We compare them to traditional mouse-driven interfaces that are commonly employed in similar contexts. The computer tools to be compared are an editor that enables to modify a sound sample on-the-fly, a sequencer that facilitates to create a score from scratch and a multi-track recorder/sampler that allows to quickly cut and mix sampled sounds.

The comparative usability study is conducted to a set of representative population. The interfaces we propose are oriented to a wide range of users, experts and novices, in the tradition of creativity support tools. The range of representative participants considered has a varied background (with or without computer literacy as well as with or without musical knowledge). Under controlled conditions, the users are observed with the aim of collecting quantitative and qualitative data, for its further analysis and discussion.

4.2.3 Evaluation criteria

The criteria to be evaluated consist in measuring the following:

- **Effectiveness**. The effectiveness is defined quantitatively by speed of performance. A comparison between pairs of systems is done through measuring quantitatively the performance of a set of representative musical tasks. This may provide quantitative data that can be compared in terms of percentage of total users.

- **Usefulness**. The usefulness (ease of use) is measured in terms of an assessment of the users motivation for using the interface and if it allows to achieve his or her goals: constraints, affordances and accuracy but also explorability and playfulness. Questions to be answered are if the interface(s) is adequate for the activity, if it invites to explore and discover, if it responds as expected or if the user could fulfill her objectives. In addition, observations about the thinking aloud will also be considered.

- **Learnability**. Learnability (easy to learn) concerns the users ability to operate the system in relation to some competence level and after some training period. It is another attribute to be measured but it is difficult to minimize bias due to the different levels of computer literacy and music knowledge among users. For each system there can be two phases: free exploration and guided exploration. Once the user knows what she can really do with the system, questions like ''could you find the options without guidance or tutorials?"

can be done in order to assess if it is easy to learn. In addition, it is important to consider those users that already know about the computer music tools evaluated such as a sound editor, a waveform sequencer or a 4-track recorder because their learning curve is different than the novel users one.

- **Experience**. The user experience is measured qualitatively according to the personal opinion about the musical sketches composed during the session, in terms of likability and satisfaction. They will be asked to rate the interfaces. That will help us in order to find possible reasons for problems that have occurred. The main questions are if the user is satisfied with the material produced, and if she would incorporate the tools(s) to her music studio in the case of the expert musician or for learning music in other case. In addition, if the experience has been perceived as positive, the musicians will probably respond afirmatively about incorporating these tools in their workflow. In order to identify interaction problems and refine the interfaces, it would be interesting to know if she misses some features and in affirmative case, which are they.

In sum, we evaluate the following hypotheses:

- **H1** At the early stages of a creative process, interfaces closer to action-driven or gestural-based interaction **are rated higher for composing music** than those mouse-driven, both for expert and novel musicians.
- **H2** The sketch-based systems for composing music are **less difficult to learn** when compared to mouse-driven interfaces.
- **H3** The sketch-based systems for composing music are **less difficult to use** than mouse-driven interfaces.
- **H4** Using sketch-based systems for composing music is an **experience more satisfactory** than using mouse-driven interfaces.
- **H5** Using sketch-based systems for composing music lead to **more effective performance** when compared to mouse-driven interfaces.

4.2.4 Set-up

The testing set-up can be carried out in a single room. In order to minimize bias, the order of the systems will vary per each participant, and also the order of the pairs. The participants will be asked to perform the musical tasks while thinking aloud. The session will be videotaped. At the end of testing the system, a questionnaire will be conducted with initial profile questions about age, gender, expertise in music and/or computer music and later on questions concerning the effectiveness, usefulness, learnability and experience using the systems. Optionally, a public talk can be initiated, in order to discuss in group those issues found.

4.3 Interfaces and tasks overview

In table 4.1, a comparison test will be carried out to evaluate the following three pairs of interfaces, which are similar in concept albeit their interaction model differ:

Tool	**Sketch-based**	**Mouse-driven**
Sound editor	*waveTable*	*Amadeus*
Sequencer	*wiiteBoard*	*Metasynth*
4-track recorder	*tr4ck*	*Ableton Live*

Table 4.1: Interfaces to be evaluated.

For that purpose, the evaluation is focused on performing similar tasks between the pairs of systems, tasks related to the music ideation process, which are the following: *sketching by editing a sound*, *sketching by drawing a sound* and *sketching by sampling sounds*.

4.3.1 Sketching by editing a sound: *Amadeus* and *waveTable*

A sound or digital audio editor is a computer application for manipulating sound samples. The interfaces to be evaluated are two sound editors: the commercial software Amadeus[1] and the waveTable prototype (§3.1). The task consists in that after loading a sound from a given library, previously selected from the collaborative database Freesound[2], the participant can modify it creatively until being happy with the musical composition. The sound should be played within a loop. The available tools for that task are the editing tools such as copy and paste, erase, as well as the effects tools, which are varied and depend on each of the programs.

[1] http://www.hairersoft.com/Amadeus.html
[2] http://www.freesound.org

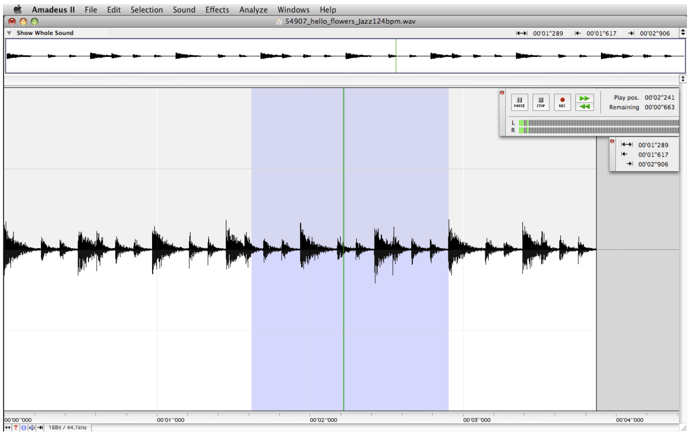

Figure 4.1: Screenshot of the Amadeus interface.

In Amadeus there is a timeline to structure the reproduction of the waveform (see figure 4.1). Amadeus represents the traditional desktop environment most musicians using computers will find to edit and modify sounds. Each operation is within an upper pop-up textual menu. An effect is applied when clicking on the *Effects* tab and selecting one of the effects available: if the selected effect would have parameters a modal window is opened which does not close until the user clicks on the *OK* or *Cancel* button. Playing in loop mode is not obvious: it has to be activated a tiny icon at the bottom left corner of the window. *Copy* and *Paste* works similarly than an office application, it can be applied using the keyboard shortcuts or via the upper menu. There is a *current time indicator* that indicates what part of the sound sample is playing and a bottom horizontal bar that indicates the duration.

Comparatively, in waveTable there is also a timeline, but it is projected onto a table (see figure 4.2). It is a text-free interface, meaning that there is not a duration metrics, only a *current time indicator*. The projected waveform of the sample can be zoomed and scrolled using finger gestures, and it is possible to modify a sound sample using tangible artifacts as well as fingers. Each operation is contained in a tangible, indicated with a printed icon. The user stands while playing, instead of sitting, enabling the collaboration. An effect is applied to the sound sample situating an effect tool onto the table, and the most prominent parameter can be modified by moving and rotating the object. The prototype only plays in loop mode. *Copy* stores a fragment by dragging onver the waveform along the x-axis and *Paste* stamps that fragment at the object position and repeats it when moved along the x-axis.

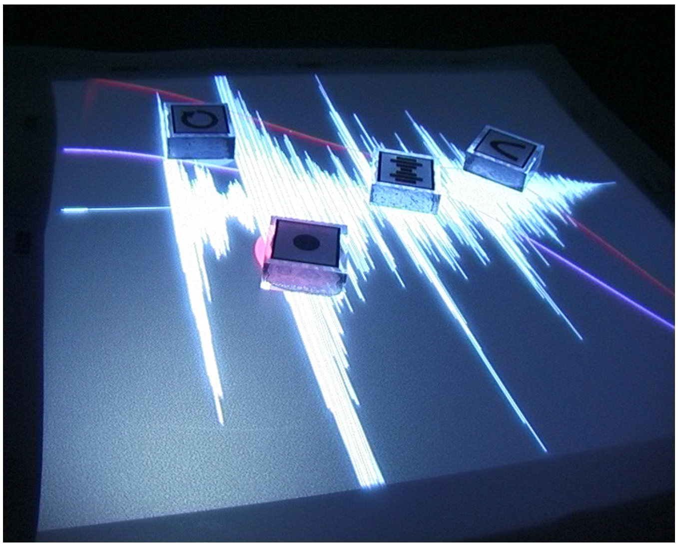

Figure 4.2: Snapshot of the waveTable interface.

4.3.2 Sketching by drawing a sound: *MetaSynth* and *wiiteBoard*

A digital music sequencer is a computer application that allows to generate computer music. The interfaces to be evaluated are two spectrographic music sequencers: the *Image synth* module of the commercial software MetaSynth[3] and the wiiteBoard prototype (§3.2). The task consists in drawing sounds on the canvas until the participant is happy with the musical composition. The sound should be played within a loop. The available tools for that task are the pencil tool.

MetaSynth is an electronic music environment conceived as a traditional desktop application (see figure 4.3). One of the six possible rooms is specialized in painting sounds on a score, the *Image synth*. The operations can be found in the upper, left or lower iconic menu. In the middle, there is a small canvas with a subtle grid where it is possible to draw from any direction to any direction. As the time indicator advances, the current notes are played, from left to right. The vertical position determines the pitch. Colour and brightness determine the spatial position and the

[3] http://www.metasynth.com

volume, respectively. It plays in loop mode when user presses the *Play* button. The user can draw points, lines or patterns. There is also a collection of types of strokes. If wanted, it is possible to modify the instrument timbre clicking on the corresponding pop-up iconic button, as well.

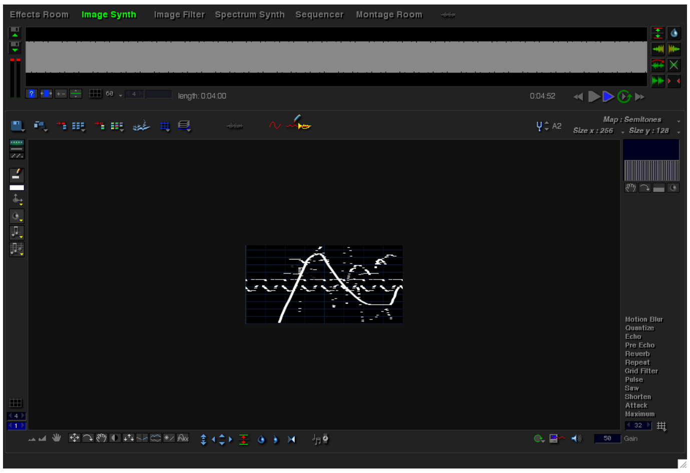

Figure 4.3: Screenshot of the Metasynth interface in the *Image synth* room.

In comparison, the wiiteBoard interface consists in a large black screen, same size than the computer screen, text-free, neither menus nor grid (see figure 4.4). There is only a *current time indicator* that moves from left to right. The user can draw straight and curved lines as well as points on the screen with the light pen of white colour, being only possible to draw from left to right. The vertical position determines the pitch. When the time indicator advances, it plays the notes founded in that precise moment and turns the strokes to blue colour. It always plays in loop mode. The instrument has one single timbre.

4.3.3 Sketching by sampling sounds: *Ableton Live* and *tr4ck*

A digital multi-track recorder is a computer application that allows to record multiple sound sources separately to later on mix them in a single output sound. Closely there is the sampler, which is an electronic music instrument that enables to play multiple recordings and manipulate them while playing. The interfaces to be evaluated are two multi-track recorders: the recording audio module of the commer-

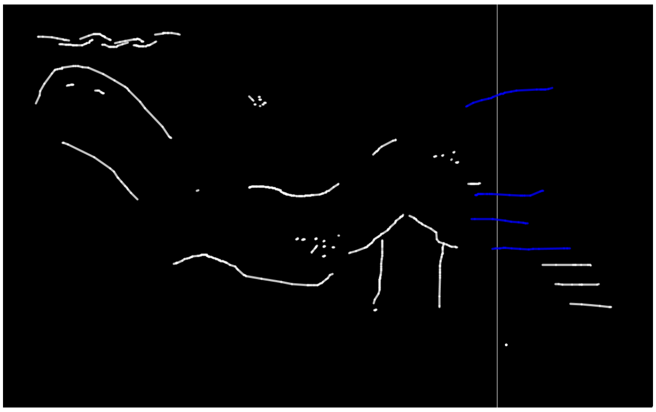

Figure 4.4: Screenshot of the wiiteBoard interface.

cial software Ableton Live[4] and the tr4ck prototype (§3.3). The task proposed is to record two to four sounds from around (one per track), and mix them until the user is happy with the musical composition. The sound should be played within a loop. The available tools for that task are the classical VCRs or CD players operations: the record and play tools, the set volume tool, and basic editing operations such as setting the loop duration.

Ableton Live is a loop-based music sequencer that serves as an instrument for live performances as well as a tool for composing, therefore it is conceived both as desktop or laptop application. *Recording an audio* is held in the *Arrangement view*, which is based on the traditional software sequencer interface of rows representing tracks (see figure 4.5). Each track has an *In/Out mixer* where the user can define the input source of the recording as well as the output target of the recording. Before recording, it is necessary to arm the track clicking on the *Arm* button. There is a level meter that indicates the volume of the sound to be recorded which ought to be adjusted. In addition, the monitoring allows to determine how to listen the output, which should be configured. In order to reset the arrangement to the beginning, the *Stop* button should be pressed twice. Then, the *Global Record* button should be pressed to announce the recording and later the *Play* button should be pressed or the *space bar*'s on the keyboard should be hit in order to start recording. During the recording, as the *current time indicator* advances, you can see how the recorded

[4] http://www.ableton.com

track is created. To stop recording, the *Stop* button should be pressed or the *space bar* hit. The recordings can be played in loop mode any time. It is possible to define in and out points of each track in order to discard parts of the sound, as well as of the general arrangement in order to control the section to be looped.

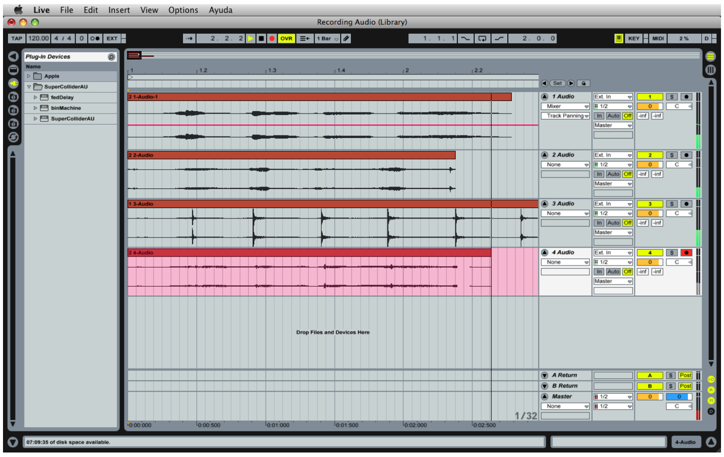

Figure 4.5: Screenshot of the Ableton Live interface when recording an audio (*Arrangement view*).

Alternatively, Live offers a less traditional way of recording tracks, held in the *Session view*, that is better suited for live performance (see figure 4.6). This view is structured as a table with cells where the columns determine the tracks and the rows determine the current state, as if it were a playlist of patterns. For start/stop recording in a selected track, it is necessary to press its lower *Arm* button, and then press the grey circle inside one cell of that track. Similarly to the Arrangement view, during the recording, you can see how the recorded track is created as the *current time indicator* advances. Once recorded, there are editing tools together with a close view of the sound sample at the bottom right.

The tr4ck interface is, if we compare to these two recording practices, conceptually closer to the *Session view* (see figure 4.7). It is conceived to be run on PDAs, then its interface is small and simple: There is a 2x2 matrix of four tracks, each one with a set of three large buttons (record, stop, play) and two large sliders, one sets the volume and the other sets the loop length. The mechanism is straightforward: To start recording, the user presses with the stylus the Record button, to stop it, the user presses with the stylus the Stop button. Similarly, to start playing, the user presses

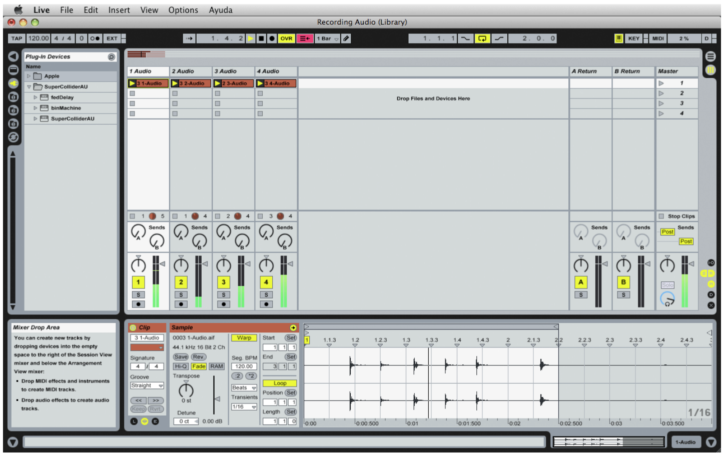

Figure 4.6: Screenshot of the Ableton Live interface when recording an audio (*Session view*).

the Play button, and to stop it, the user presses stop again. When mixing, meaning that more than one track is in loop-based playing mode, it is possible to activate the loop length and determine a shorter sample duration that always starts at the same in point. In addition, the volume of each track can be modified.

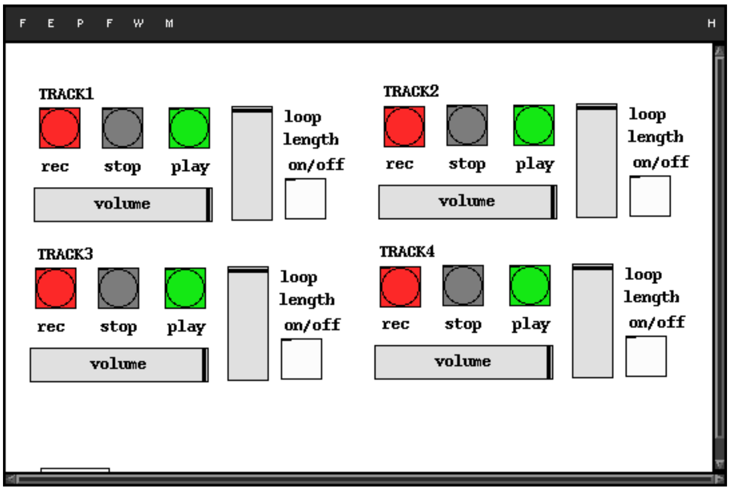

Figure 4.7: Screenshot of the tr4ck interface.

Chapter 5

Analysis and Discussion

In this chapter we first introduce the context of the usability study conducted. Second, we analyse the overall user ratings. Next, we highlight the feedback obtained from the think aloud and interview data and concrete the results of the learnability, usefulness, experience and performance aspects. Finally, we discuss what do we proved and what do the results illustrate.

5.1 Introduction

The usability study has been carefully conducted during 4 days with several representational users (see Appendix A). In total, 8 participants have been recruited (2 female and 6 male). The participant mean was 33 years old, with less than two years of musical training and with a basic computer music knowledge of having tried some music program(s). The usability tests of the editor and the sequencer were conducted during two days in a lab. Similarly, the 4-multitrack recorder study was conducted during another two days under mobile conditions. All participants were asked to perform a concrete task trying one, two or three pairs of systems (§4.3). For the analysis, we have considered *beginners* in making music with computers those that have never tried or have tried some program(s) (75%) while the *advanced* those that make music occasionally or regularly (25%).

5.2 User ratings of the UI designs

Our first hypothesis (§4.2.3) was that *at the early stages of a creative process, interfaces closer to action-driven or gestural-based interaction* ***are rated higher for composing music*** *than those mouse-driven, both for expert and novel musicians*. We found that this was the case for both groups.

Each prototype within the questionnaire was rated on 8 measures: one ease-of-learning rating, four ease-of-use ratings and three like/dislike experience ratings. We have considered all of them in order to find what interface has been rated higher. The results have been divided into beginners, advanced and overall results.

As shown in figure 5.1, beginners rated higher the sketch-based sequencer (wiiteBoard, and the Ableton Live multi-track recorder was the worst of the six. Advanced rated higher the portable tr4ck while the mouse-driven MetaSynth sequencer was the worst rated (see figure 5.2). Finally, the overall ratings counting both groups point out that the sketch-based sequencer (wiiteBoard) is the prototype higher rated, whilst the traditional editor Amadeus is the worst rated (see figure 5.3 and table 5.1). In all three cases the sketch-based interfaces obtain higher ratings than the traditional ones.

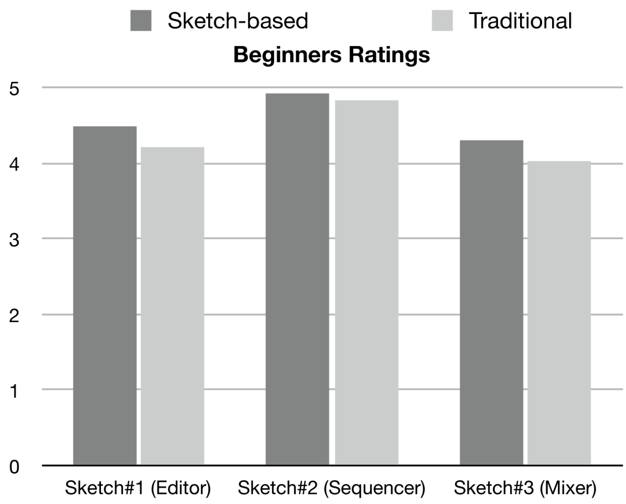

Figure 5.1: Mean of *Beginners* overall ratings *(1-5)* for the six designs.

5.3 Feedback from the think aloud and interview data

The verbal feedback regarding think-aloud videotaped during the task performance and interview data talking about the filled comment fields of the questionnaire was analyzed.

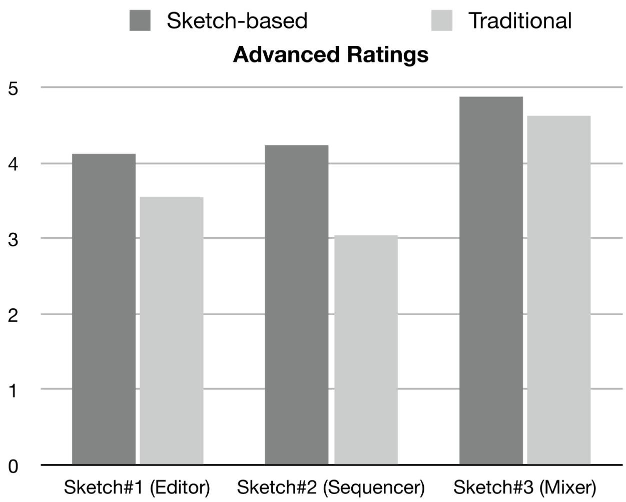

Figure 5.2: Mean of *Advanced* overall ratings *(1-5)* for the six designs.

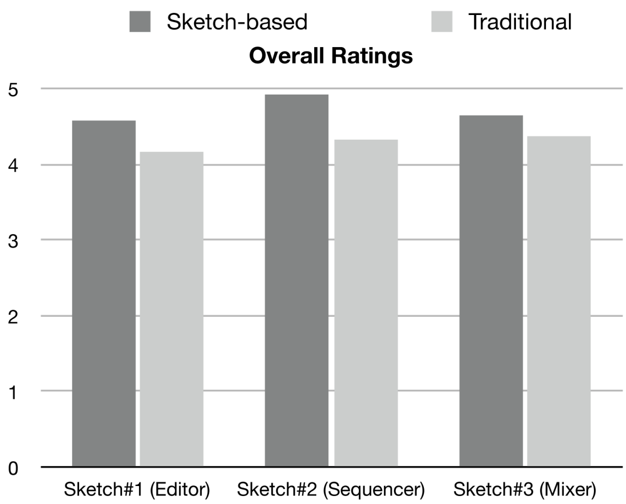

Figure 5.3: Mean of overall ratings *(1-5)* for the six designs (*Beginners* and *Advanced*).

	Sketch#1 (Editor)	Sketch#2 (Sequencer)	Sketch#3 (Mixer)
Sketch-based	4.13 *(SD=1.14)*	**4.43** *(SD=0.74)*	4.19 *(SD=0.77)*
Traditional	3.76 *(SD=1.22)*	3.90 *(SD=1.28)*	3.94 *(SD=0.93)*

Table 5.1: Mean and (*standard deviation*) of overall ratings *(1-5)* for the six designs (*Beginners* and *Advanced*).

Our second hypothesis (§4.2.3) was that the *sketch-based systems for composing music are* ***less difficult to learn*** *when compared to mouse-driven interfaces*. This turn out not to be the case neither for beginners nor for advanced users.

If we observe the learnability rating (see figure 5.4), beginners rated the sketch-based system waveTable as the most difficult to learn. Advanced rated first MetaSynth and then waveTable with the lower ratings. In sum, waveTable is seen as the most difficult system. In general, beginners and advanced asked for assistance about the objects significance, and beginners in particular reclaimed a tutorial and time to understand better the instrument. However, for beginners the interface that was better rated was MetaSynth with opinions that ranged from "You obtain pleasant sound immediately" to "No assistance was required". Advanced users, at the contrary, rated ease-to-learn the half of the six systems: wiiteBoard, Live and tr4ck, and curiously left the MetaSynth in the last position because "it offers hundreds of options". From the overall ratings we obtain that the sketch-based wiiteBoard interface is rated as the more easy to learn and the sketch-based waveTable is rather rated as the most difficult.

Our third hypothesis (§4.2.3) was that the *sketch-based systems for composing music are* ***less difficult to use*** *than mouse-driven interfaces. We found that this was the case for both groups*.

In general, participants rated tr4ck as the higher easy to use saying that "it extremely invites you to explore because it is portable", "it invites you to explore what you can do with recordings in the environment", although "you cannot follow the rhythm". Advanced rated next Live, commenting that "it is more complicated for exploring" but "it is very usable". Beginners, rather, rated higher wiiteBoard commenting about "how easy it is to experiment", and then the similar MetaSynth that is considered as well as "ease-to-experiment". In the same position we find waveTable, that "invites you to explore because it is more interactive", "it has more

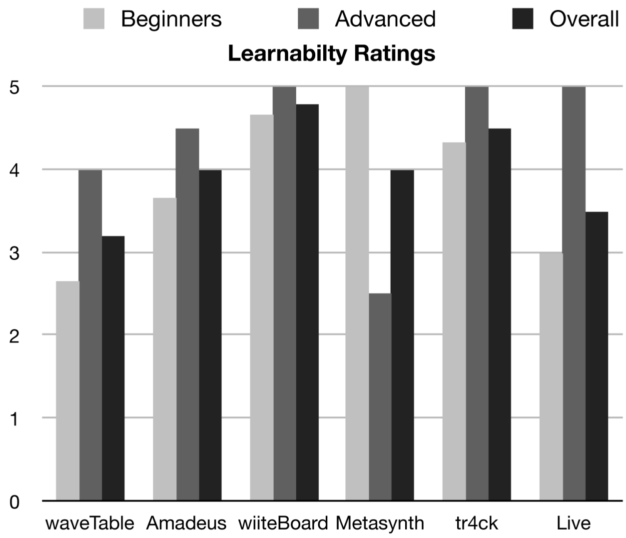

Figure 5.4: Mean of learning rating for the six designs *(1 very difficult - 5 very easy)*.

potential", "it correlates a lot of senses: moving the objects with touch, sound and vision" and "it fits well with performance live". Similarly, advanced users think that "it is ideal for artistic exhibitions because of its size" although not all of them would incorporate the tool in their studio.

As we can observe in diagram 5.8, there are more comments on the positive aspects of interfaces for sketching musical compositions than for similar tools more devoted to point-and-click. These positive aspects are counteract with much more criticism about the interface accuracy.

Regarding the *affordances* and *constraints* of the interfaces evaluated, according to the overall usefulness ratings in figure 5.5 , the Amadeus editor has the lower rating, "it is more offline", and a participant with few computer literacy affirms that "the mouse produce me claustrophobia", while others hesitate "what to do" and "what to explore". However, the waveTable editor affords to zoom in and out with the fingers and explore, but then "you loose the global reference". It also invites to collaborate between people and explore together. In the case of the sequencers, users prefer wiiteBoard to MetaSynth in overall usefulness ratings terms, however, pressing the wiiteBoard's light pen button to activate drawing is not obvious, and

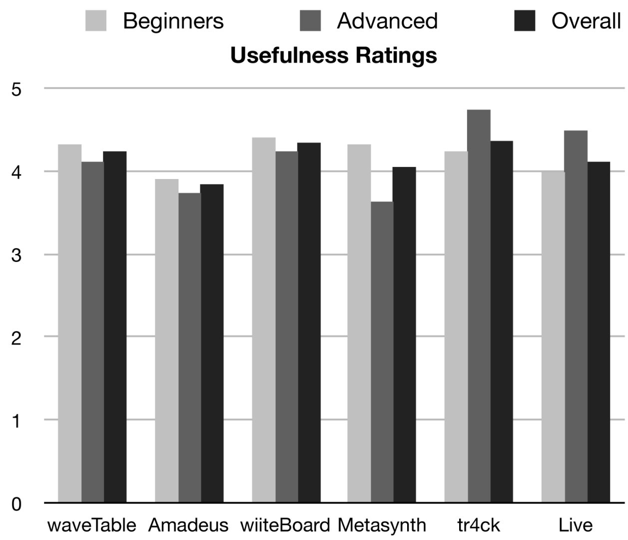

Figure 5.5: Mean of usefulness ratings for the six designs *(1 completely disagree - 5 completely agree)*.

during the painting task users try to draw real objects like houses or hearts with no success due to its few accuracy: "if it responded more accurately, it would be better than the mouse". At the contrary, MetaSynth is easy to be controlled using the mouse, but the window-canvas for drawing is small and "I only can create notes and not continuous lines", although when finishing the sketch, some users treated the sketch as a pictorial masterpiece: "where do I sign?". Finally, Live is seen as tool for live performance as well as for desktop production, and the general affordance is a controllable but complex tool. Using tr4ck, the users did more trial-and-error recordings, and tended to explore more the sonic environment, sometimes using the stylus as a source of sound.

Our fourth hypothesis (§4.2.3) affirmed that *using sketch-based systems for composing music is an **experience more satisfactory** than using mouse-driven interfaces*. We realized that this was the case for both, beginners and advanced, but not for the overall experience ratings.

Let's analyze each of the groups: while the three interfaces that beginners rated higher were, in sequential order: wiiteBoard, MetaSynth and waveTable; advanced users rated higher the following: tr4ck, Live and waveTable. In general, they sug-

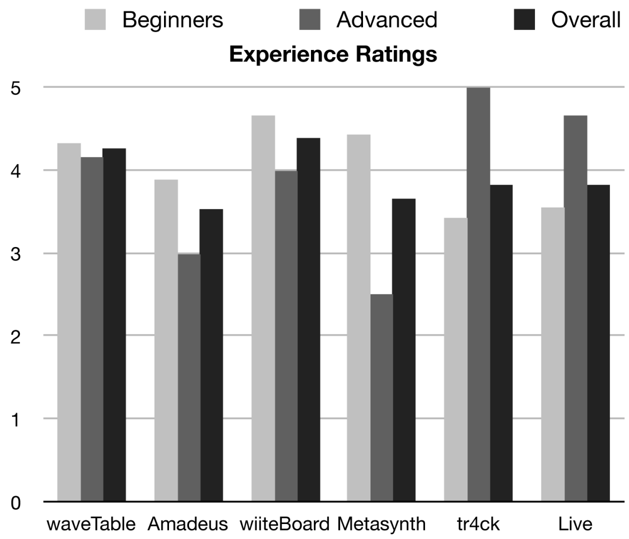

Figure 5.6: Mean of experience ratings for the six designs *(1 completely disagree - 5 completely agree)*.

gested more functionalities in all the interfaces: ”more effects”, ”crop sounds”, ”undo”, ”drawing precision” in the case of the waveTable editor; ”a toolbar”, ”presets”, ”filter visual feedback” in the case of the Amadeus editor; ”undo”, ”erase”, ”drawing precision”, ”grid” and ”frequency indicator” in the case of the wiiteBoard sequencer; ”continuos pitch”, ”drawing precision”, ”erase” and ”frequency indicator” in the case of the MetaSynth sequencer; ”loop start control”; ”greater portability” in the case of Live and, finally, ”crop sounds” and ”velocity” in the case of tr4ck.

5.4 Analysis of sketches performance

Our fifth hypothesis (§4.2.3), which says that *using sketch-based systems for composing music lead to* ***more effective performance*** *when compared to mouse-driven interfaces*, measuring effective performance in terms of task timings and task accuracy, it turns out not to be the case.

After measuring the raw data of users task timings, we obtain the following (see figure 5.2):

#User ID	Time for Sketch#1 (m)		Time for Sketch#2 (m)		Time for Sketch#3 (m)	
	Sketch-based	Traditional	Sketch-based	Traditional	Sketch-based	Traditional
1	2.3	1.7	1.4	2.4	4.2	2.4
2	11.9	6.9	6.4	5.2	-	-
3	2.0	1.8	3.4	0.5	-	-
4	4.3	11.8	2.6	1.3	-	-
5	4.8	9.9	1.6	0.5	-	-
6	-	-	-	-	2.8	5.6
7	-	-	-	-	4.5	6.7
8	-	-	-	-	8.72	5

Table 5.2: Composing time for different sketches using different UIs.

Due to the mean time is a coarse indicator of the whole performance, we calculate instead the median time (see 5.7). Thus, the overall median time to complete the sequencing task is shorter in the case of the point-and-click application (1.3 minutes) compared to the sketching application (2.6 minutes). Similarly, the mixing and sampling task is shorter in the case of the point-and-click application (4.4 minutes) than in case of the sketching tool (5.3 minutes). However, the editing task has been completed in shorter time within the sketching application (4.3 minutes) comparing to the point-and-click tool (6.9 minutes).

If we observe each of the two groups, we notice that the behaviour explained above happens in a similar way. However, in the advanced group there are slight differences: the median time for the editing task is shorter in the case of the mouse-driven application (1.8 minutes) compared to the sketching one (7.1 minutes) and the median time for the mixing and sampling task is shorter with the sketching tool (2.4 minutes) than for the desktop application one (4.2 minutes).

5.5 Discussion

This study illustrates how participants, both beginners and advanced, rated higher interfaces for sketching musical compositions (an editor, a sequencer and a sampler), when compared to mouse-driven interfaces. Participants sometimes felt uncomfortable with the former systems when the input device did not work with accuracy, but the thinking aloud and later comments were more extensive about creative ideas.

Learnability is an issue that depends on the user background. In addition, not knowing the interaction model can be an impediment for rapid ideation in a short

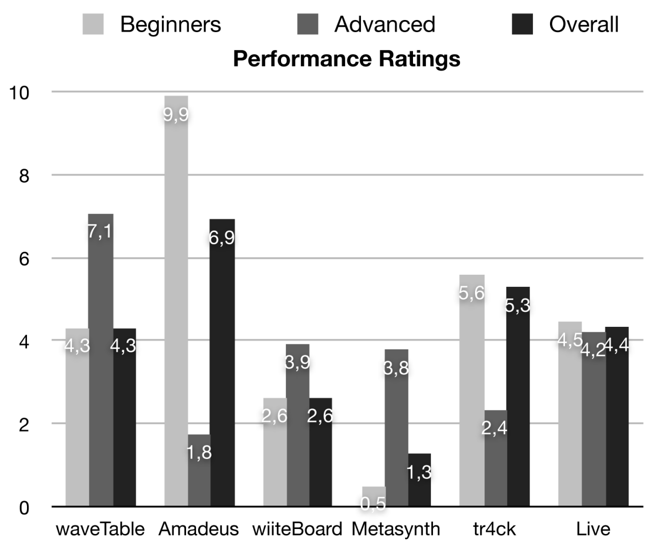

Figure 5.7: Median time (m) to complete the six tasks.

period of time. Advanced in computer music rated higher the UI that they already knew (e.g. Live or Amadeus) or the UI easy to grasp (e.g. wiiteBoard) independently of knowing the tool (e.g. MetaSynth). Beginners rated higher simple concepts and with visual feedback (e.g. wiiteBoard, MetaSynth or tr4ck), feeling more comfortable if they already knew the interaction model. For a person with low computer literacy, it has been surprising her preference for sketch-based interfaces.

Regarding *usefulness*, in general participants consider sketch-based tools more useful when compared to those mouse-driven, indenpendtly of the difficulty of the interaction model. In particular, the advanced group prefer those multi-functional and flexible tools (e.g. Ableton Live, tr4ck or waveTable), whilst beginners prefer the interfaces with more visual feedback, although they don't understand the whole functioning (e.g. waveTable).

Summarizing the *experience* of users, they agree that the sketching tools have potential, but they would need much more time to really learn how to use and enjoy them. While some beginners were satisfied with the results obtained whether or not they previously knew the interaction model, advanced were more critical of the results obtained when they could not control the interface, and some beginners were unsatisfied with the overall results. Some beginners noticed that all the tools were creative, not associating creativity to the input device employed. Finally, observing the video tapes we can conclude that the techniques employed while using sketch-based interfaces have been more exploratory than when using those mouse-controlled (e.g. moving objects on waveTable to apply an effect versus point-and-clicking a filter in Amadeus, drumming with the stylus on a surface with the tr4ck recorder versus humming while seated with Live or contemplative drawing with wiiteBoard while point-and-clicking notes with MetaSynth). In addition, the frequency of creative comments and tags is higher in the case of sketch-based interfaces.

Concerning *performance*, despite calculating the time to complete each task, which is a common measure in usability testing, we hesitate if it makes sense to quantitatively measure creativity tasks. Maybe a participant devotes much more time because she is exploring how the system works, but maybe she is enjoying the activity so much that she prefers to keep exploring. We think that measuring the performance time in creative tasks can become not that relevant.

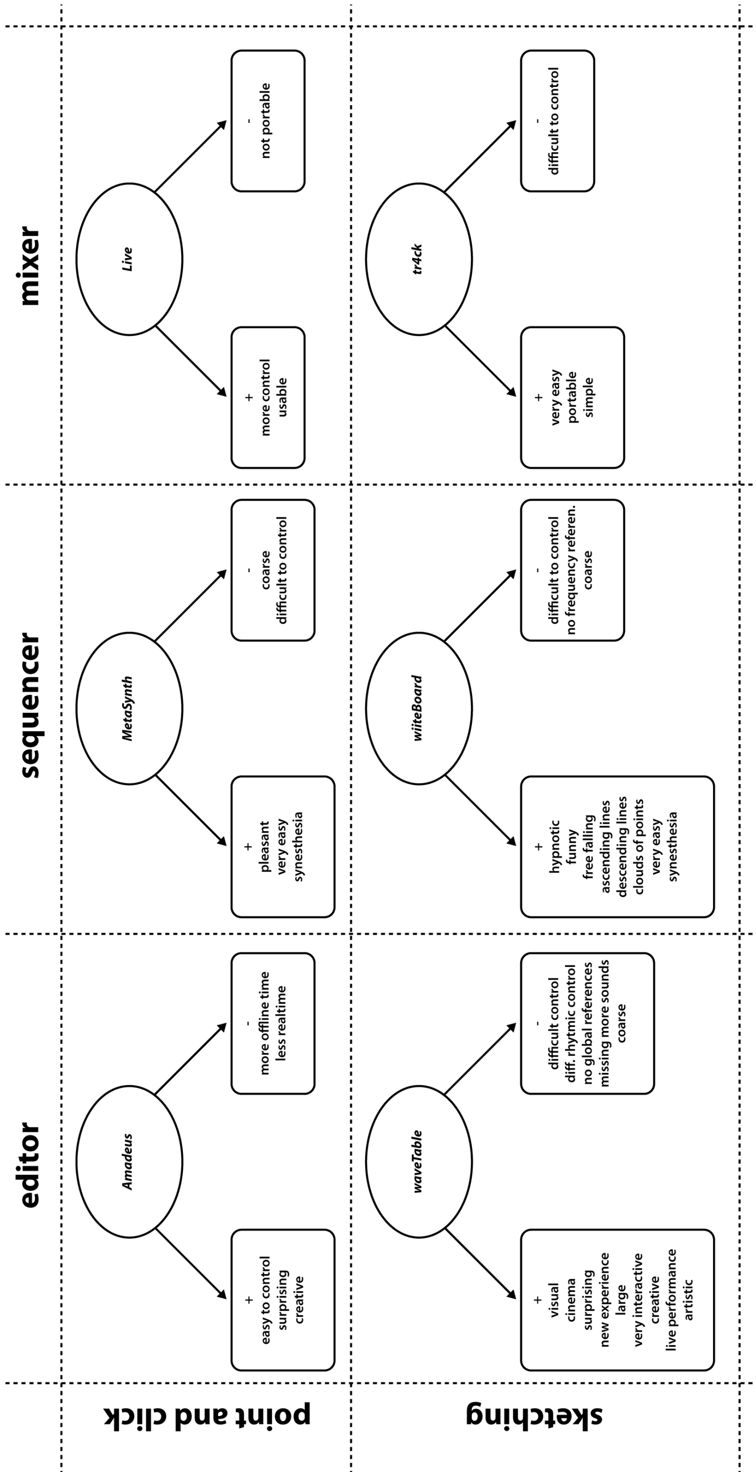

Figure 5.8: Diagram of frequent comments on the positive and negative aspects of interfaces evaluated.

Chapter 6

Conclusions

In this chapter, we summarize the contributions of this thesis. Then, we describe the open issues that have emerged proposing future research topics.

6.1 Summary of contributions

As we indicated in the introduction, the motivation of this thesis is exploring the possibilities of action-driven and stylus-based technologies to enhance musical creativity given their adequacy to embodied and creative tasks such as sketching musical compositions.

We have reviewed (of course, only partially) the more relevant aspects of the state of the art related to our goal, namely:

- tools supporting creativity, and, specifically, interactive sketching
- basic interaction design concepts
- models of computer music creativity
- evaluation of interfaces for music expression

A more particular focus has been the exploration of alternative interaction models to the traditional point-and-click, in particular, tangible, pen-driven and mobile types of interaction, which have been reviewed and characterised more thoroughly.

We have explored these interaction models through building (and evaluating) three prototypes supporting musical creativity and exemplifying each type of interaction. First, tangible interaction is looked into through the waveTable prototype, an audio waveform editor that can be operated in real time through a tabletop interface. As a musical table with tangibles, it directly deals with tangible sound editing, an issue tangentially addressed by similar systems more focused on sound generation and real-time sequencing. Preliminary evaluation indicates that the portability of the prototype and the physical objects mapping are open design issues to be improved. Second, pen-driven interaction is examined through the wiiteBoard prototype, a sequencer with gestural-driven interface that allows to paint notes on a screen using a light pen. This prototype addresses direct manipulation of sound using a light pen on a screen instead of the traditional mouse or even the pen-tablet commonly used in these applications, providing a closer environment to the mental model. Initial lessons indicate that the input device performance as well as the representation of frequency and time should be improved. Third, mobile interaction is explored through the tr4ck prototype, a mobile multitrack recorder and sampler that allows to record and mix up to four tracks simultaneously using a handheld device. This prototype addresses the possibility of recording and mixing everywhere and in few seconds, two tasks that handheld devices usually address separately. Preliminary observations inform us that performance including the quality of the recorded sound ought to be improved together with the addition of more editing tools.

A first usability evaluation, both qualitative and quantitative, comparing the prototypes to software with more traditional interfaces, has been undertaken resulting in the general preference for sketch-based systems for music ideation. The evaluation criteria considered have been learnability, usefulness, experience and performance. The results of the evaluation have illustrated that learnability is an aspect difficult to be measured in a single usability lab. In general, those systems that users did not know previously or those with perceptually much more functionalities have resulted in those interfaces most difficult to learn, however systems with visual feedback have provided confidence to users. Regarding usefulness, we have observed that users have rated higher all the sketch-based interfaces. Similarly, experience has been more exploratory using the sketch-based prototypes although in both types of interfaces users adopt a creative attitude. Finally, performance measurements have not provided relevant information, a result to be considered when measuring creative tasks.

The literature reviewed confirms that interfaces for sketching musical systems is a topic little addressed, although they can play a relevant role for enhancing creativity. We have proven that, employing HCI techniques and from a user-centered viewpoint, interfaces for sketching musical compositions are better suited for the early stages of the creative process when compared to mouse-driven desktop tools. Also, we have validated that non-computer literate participants prefer interfaces closer to the paper-and-pencil affordance when compared to interfaces mouse-controlled. In addition, we have validated that sketch-based interfaces invite non musicians to sketch compositions from the first moment.

6.2 Open issues and future work

An open issue is that musical creative tasks are mostly supported by mouse-controlled interfaces although it has been verified that users prefer alternative interaction models more sketch-based, closer to the paper affordance. Manufacturers should see enough commercial possibilities in order to start building standalone sketch-based software (and even hardware). Current mouse-driven applications are generally oriented to tasks that require precision. Building systems for music ideation using alternative interaction models should not mean loosing the precision provided by mouse-controlled systems, though. Paradoxically, stylus-based interaction is previous to the mouse-driven one, and although being demonstrated that the former works well for the creative process, it is used not that much for creative tasks. One possible reason is the lack of stylus precision. For instance, the PDA's stylus is still not enough precise for certain activities such as computer music patching (an activity close to sketching). The advent of tangible and multi-touch interaction has bodily reinforced creative activities. We have observed that it is a promising field that enhances activities computer-supported in a way completely different when compared to the desktop computer environment, however, its novelty causes that, currently, building low-cost prototypes with audiovisual feedback still imply awkward set-ups, and depending on the components used, the budget increases up to being difficult to be commercialized. However, the positive results obtained in this thesis indicate that more interfaces for supporting musical sketching based on stylus, tangible or mobile interaction should be explored.

Another open issue concerns the evaluation methodology for musical creativity support tools in order to evaluate creativity enhancement. We have confirmed that some information regarding the experience using the prototypes is missed when conducting a usability lab. Techniques for conducting longer and periodic usability studies of these systems should be defined (comparative or standalone). For instance, a long-period usability test could be conducted in order to observe if users find differences between sketch-based systems and similar mouse-controlled systems. Similarly, evaluation of the user experience as well as of the learning curve using creativity support tools for music ideation should be measured in the long term. Measuring creativity can be done according to Csikszentmihalyi's Flow Theory, cited in (Pachet 2006). This theory is based on the connection between challenges and skills when facing a creative activity, that ideally can produce the state of Flow: high challenges and low skills produce anxiety that with more training can become high challenges and high skills, namely Flow. Although waveTable has been rated in the usability study as the most difficult tool to be learned, we have observed a rapid learning curve among users, beginners and advanced. A long-term observation would be interesting to carry out, also exploiting the collaborative aspect. In the case of mobile interaction prototypes such as tr4ck, the technique explained by (Brandt et al. 2007), consisting in a diary study carried out under mobile conditions can be applied.

Other evaluation techniques could be focused on the sketches themselves: sketches could be analyzed quantitatively and by content using clustering techniques such as those employed in (Tohidi et al. 2006). In addition, alternative techniques more based on signal processing treating sketches as time series could be analysed. Given the importance of the proper sketching activity, and the physical action itself, deeper studies of the performer movements using movement visualization techniques could add relevant information, like the *motiongrams* visualization described in (Jensenius 2007).

Also, the UI design of the prototypes should be improved maintaining the use of low-cost components. According to the usability study conducted, we should deal with a compromise between accuracy and unconstraint when designing interfaces for sketching musical compositions, in order to maintain an adequate creative environment but also allowing user interface control. More usability tests of the iterative prototypes should be conducted as well. Concretely, the UI design and mappings of waveTable should be improved in terms of solving the current enigmatic meaning of the phicons perceived by users, as well as reducing the size of the prototype and improving portability for testing in live performance settings. In the case of the accuracy of the wiiteBoard's light pen, it should be improved in terms of mapping better the drawing activity. Mobile recording should be complemented with more editing tools available *in situ*, exploiting much more the stylus-based interaction of the device. In general, an integration among these tools, and also with high-end tools would make sense to support the whole creative process exposed at the beginning of this thesis.

In addition, given the proved ease-of-use of the interfaces for sketching musical compositions, it should be fruitful to conduct some usability tests within the educational field in order to obtain some feedback from both children and teachers of computer music.

Creativity support tools for musical ideation are a little bit constrained by the point-and-click model of interaction. This thesis takes steps toward enhancing creativity using both the capacity of computer systems together with the richness and variety of the interaction with the physical world.

Bibliography

Adler, A., Eisenstein, J., Oltmans, M., Guttentag, L. and Davis, R.: 2004, Building the design studio of the future, *Making Pen-Based Interaction Intelligent and Natural*, AAAI Press, Menlo Park, California, pp. 1–7.

Barthelmes, B.: 2002, Music and the city, *in* H.-J. Braun (ed.), *Music and Technology in the Twentieth Century*, pp. 97 – 105.

Bau, O., Tanaka, A. and Mackay, W. E.: 2008, The A20: Musical metaphors for interface design, *in* A. Camurri, S. Serafin and G. Volpe (eds), *NIME'08: Proceedings of 2008 conference on New interfaces for musical expression*, Genova, Italy, pp. 91–96.

Berry, R., Makino, M., Hikawa, N. and Suzuki, M.: 2003, The Augmented Composer Project: The Music Table, *ISMAR '03: Proceedings of the 2nd IEEE/ACM International Symposium on Mixed and Augmented Reality*, IEEE Computer Society, Washington, DC, USA, p. 338.

Blackwell, A. F.: 2006, The reification of metaphor as a design tool, *ACM Trans. Comput.-Hum. Interact.* **13**(4), 490–530.

Blackwell, A. F., Bailey, G., Budvytis, I., Chen, V., Church, L., Dubuc, L., Edge, D., Linnap, M., Naudziunas, V. and Warrington, H.: 2007, Tangible interaction in a mobile context tangible interaction in a mobile context tangible interaction in a mobile context, *CHI 2007*.

Boll, S.: 2005, Image and video retrieval from a user-centered mobile multimedia perspective, *CIVR*, pp. 18–27.

Bongers, B.: 2006, *Interactivation – towards an e-cology of people, our technological environment, and the arts*, PhD thesis, Vrije Universiteit Amsterdam.

Brandt, J., Weiss, N. and Klemmer, S. R.: 2007, txt 4 l8r: Lowering the burden for diary studies under mobile conditions, *CHI '07: CHI '07 extended abstracts on Human factors in computing systems*, ACM, New York, NY, USA, pp. 2303–2308.

Buxton, B.: 2007, *Sketching user experiences: Getting the design right and the right design*, first edn, Morgan Kaufmann.

Buxton, W.: 1995, Touch, gesture and marking, *in* R. M. Baecker, J. Grudin, W. Buxton and S. Greenberg (eds), *Readings in Human-Computer Interaction: Toward the year 2000*, Morgan Kaufmann Publishers, pp. 469 – 524.

Cooper, A., Reimann, R. and Cronin, D.: 2007a, Implementation models and mental models, *About Face 3: The Essentials of Interaction Design*, Wiley Publishing, pp. 27 – 40.

Cooper, A., Reimann, R. and Cronin, D.: 2007b, Understanding users: Qualitative research, *About Face 3: The Essentials of Interaction Design*, Wiley Publishing, pp. 49 – 73.

Costanza, E., Shelley, S. and Robinson, J.: 2003, Introducing Audio d-touch: a novel tangible user interface for music composition and performance, *Proceedings of the 6 th international conference on digital audio effects (DAFx-03)*, London.

Coughlan, T. and Johnson, P.: 2006, Interaction in creative tasks: Ideation, representation and evaluation in composition, *CHI '06: Proceedings of the SIGCHI conference on Human Factors in computing systems*, ACM, New York, NY, USA, pp. 531–540.

Cox, C. and Warner, D.: 2004, Dj culture, *in* C. Cox and D. Warner (eds), *Audio Culture: Readings in Modern Music*, Continuum, New York, pp. 329–330.

de la Motte-Habber, H.: 2002, Soundsampling: An aesthetic challenge, *in* H.-J. Braun (ed.), *Music and Technology in the Twentieth Century*, pp. 199 – 206.

Dickinson, J. K., Yu, Z., Zeng, Y. and Antunes, H.: 2005, Pen-tablet as a CAD interface alternative, *Robotics and Computer-Integrated Manufacturing* **21**(4-5), 465–474.

Dourish, P.: 2001, *Where the action is: the foundations of embodied interaction*, MIT Press, Cambridge, MA, USA.

Farbood, M. M., Pasztor, E. and Jennings, K.: 2004, Hyperscore: A graphical sketchpad for novice composers, *IEEE Comput. Graph. Appl.* **24**(1), 50–54.

Fernaeus, Y., Tholander, J. and Jonsson, M.: 2008, Towards a new set of ideals: Consequences of the practice turn in tangible interaction, *TEI '08: Proceedings of the 2nd international conference on Tangible and embedded interaction*, ACM, New York, NY, USA, pp. 223–230.

Fishkin, K. P.: 2004, A taxonomy for and analysis of tangible interfaces, *Personal and Ubiquitous Computing* **8**(5), 347–358.

Gadd, A. and Fels, S.: 2002, Metamuse: metaphors for expressive instruments, *NIME '02: Proceedings of the 2002 conference on New interfaces for musical expression*, National University of Singapore, Singapore, Singapore, pp. 1–6.

Geiger, G.: 2003, PDa: Real time signal processing and sound generation on handheld devices, *Proceedings of the 2003 International Computer Music Conference*, International Computer Music Association.

Geiger, G.: 2006, Using the touch screen as a controller for portable computer music instruments, *NIME '06: Proceedings of the 2006 conference on New interfaces for musical expression*, IRCAM — Centre Pompidou, Paris, France, France, pp. 61–64.

Goina, M. and Polotti, P.: 2008, Elementary gestalts for gesture sonification, *in* A. Camurri, S. Serafin and G. Volpe (eds), *NIME'08: Proceedings of 2008 conference on New interfaces for musical expression*, Genova, Italy, pp. 150–153.

Healey, P. G. T. and Thiebaut, J. B.: 2007, Sketching musical compositions, *Proceedings of Cogsci 2007: 29th Annual Conference of the Cognitive Science Society*, pp. 1079–1084.

Hornecker, E. and Buur, J.: 2006, Getting a grip on tangible interaction: A framework on physical space and social interaction, *CHI '06: Proceedings of the SIGCHI conference on Human Factors in computing systems*, ACM Press, New York, NY, USA, pp. 437–446.

Hunt, A., Wanderley, M. M. and Paradis, M.: 2002, The importance of parameter mapping in electronic instrument design, *NIME '02: Proceedings of the 2002 conference on New interfaces for musical expression*, National University of Singapore, Singapore, Singapore, pp. 1–6.

Jensenius, A. R.: 2007, *Action-Sound. Developing Methods and Tools to Study Music-Related Body Movement.*, PhD thesis, University of Oslo.

Jordà, S.: 2005, *Digital Lutherie: Crafting musical computers for new musics' performance and improvisation*, PhD thesis, Universitat Pompeu Fabra.

Jordà, S. and Alonso, M.: 2006, Mary had a little scoretable or the reactable goes melodic, *NIME '06: Proceedings of the 2006 conference on New interfaces for musical expression*, IRCAM — Centre Pompidou, Paris, France, France, pp. 208–211.

Jordà, S., Geiger, G., Alonso, M. and Kaltenbrunner, M.: 2007, The reacTable: Exploring the synergy between live music performance and tabletop tangible interfaces, *TEI '07: Proceedings of the 1st international conference on Tangible and embedded interaction*, ACM, New York, NY, USA, pp. 139–146.

Jordà, S., Kaltenbrunner, M., Geiger, G. and Bencina, R.: 2005, The reacTable*, *Proceedings of the International Computer Music Conference (ICMC 2005)*, Barcelona, Spain.

Kaltenbrunner, M. and Bencina, R.: 2007, reacTIVision: a computer-vision framework for table-based tangible interaction, *TEI '07: Proceedings of the 1st international conference on Tangible and embedded interaction*, ACM, New York, NY, USA, pp. 69–74.

Kaltenbrunner, M., Jordà, S., Geiger, G. and Alonso, M.: 2006, The reactable*: A collaborative musical instrument, *Enabling Technologies: Infrastructure for Collaborative Enterprises, 2006. WETICE* **0**, 406–411.

Kiefer, C., Collins, N. and Fitzpatrick, G.: 2008, Methodology for evaluating musical controllers: A case study, *in* A. Camurri, S. Serafin and G. Volpe (eds), *NIME'08: Proceedings of 2008 conference on New interfaces for musical expression*, Genova, Italy, pp. 87–90.

Klee, P.: 1968, *Pedagogical Sketchbook*, 1968 edn, Faber & Faber.

Klemmer, S. R.: 2004, *Tangible user interface input: Tools and techniques*, PhD thesis, University of California, Berkeley, Berkeley, CA, USA. Chair-James Anthony Landay.

Klemmer, S. R., Hartmann, B. and Takayama, L.: 2006, How bodies matter: five themes for interaction design, *DIS '06: Proceedings of the 6th conference on Designing Interactive systems*, ACM, New York, NY, USA, pp. 140–149.

Landay, J. A. and Myers, B. A.: 1995, Interactive sketching for the early stages of user interface design, *CHI '95: Proceedings of the SIGCHI conference on Human factors in computing systems*, ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, pp. 43–50.

Landay, J. A. and Myers, B. A.: 2001, Sketching interfaces: toward more human interface design, *Computer* **34**(3), 56–64.

Levin, G.: 2000, *Painterly interfaces for audiovisual performance*, PhD thesis, MIT.

Levin, G.: 2006, The table is the score: An augmented-reality interface for realtime, tangible, spectrographic performance, *Proceedings of the International Conference on Computer Music 2006 (ICMC'06)*, New Orleans.

Magnusson, T.: 2006, Screen-based musical interfaces as semiotic machines, *NIME '06: Proceedings of the 2006 conference on New interfaces for musical expression*, IRCAM — Centre Pompidou, Paris, France, France, pp. 162–167.

Marino, G., Serra, M.-H. and Raczinski, J.-M.: 1993, The UPIC system: Origins and innovations, *Perspectives of New Music*, Vol. 31(1), pp. 258–269.

McCartney, J.: 2002, Rethinking the computer music language: Supercollider, *Comput. Music J.* **26**(4), 61–68.

Miranda, E. R.: 2001, Computer music: facing the facts, *in* F. Rumsey (ed.), *Composing Music with Computers (with CD-ROM)*, Focal Press, Newton, MA, USA, pp. 1–20.

Moholy-Nagy, L.: 2004, Production-reproduction: Potentialities of the phonograph (1923), *in* C. Cox and D. Warner (eds), *Audio Culture: Readings in Modern Music*, Continuum, New York, pp. 331–333.

Nakakoji, K., Tanaka, A. and Fallman, D.: 2006, "Sketching" nurturing creativity: commonalities in art, design, engineering and research, *CHI '06: CHI '06 extended abstracts on Human factors in computing systems*, ACM, New York, NY, USA, pp. 1715–1718.

Nielsen, J.: 1993, *Usability engineering*, AP Professional, Boston, USA.

Nishibori, Y. and Iwai, T.: 2006, TENORI-ON, *NIME '06: Proceedings of the 2006 conference on New interfaces for musical expression*, IRCAM — Centre Pompidou, Paris, France, pp. 172–175.

Norman, D. A.: 2002, *The design of everyday things*, Basic Books.

Orio, N., Schnell, N. and Wanderley, M. M.: 2001, Input devices for musical expression: borrowing tools from HCI, *in* I. Poupyrev, M. J. Lyons, S. Fels and T. Blaine (eds), *NIME '01: Proceedings of the 2001 conference on New interfaces for musical expression*, National University of Singapore, Singapore, Singapore, pp. 1–4.

Pachet, F.: 2006, Creativity studies and musical interaction, *in* I. Deliège and G. Wiggins (eds), *Musical Creativity*, Psychology Press.

Patten, J., Recht, B. and Ishii, H.: 2002, Audiopad: a tag-based interface for musical performance, *NIME '02: Proceedings of the 2002 conference on New interfaces for musical expression*, National University of Singapore, Singapore, Singapore, pp. 1–6.

Poupyrev, I., Lyons, M. J., Fels, S. and (Bean), T. B.: 2001, New interfaces for musical expression, *CHI '01: CHI '01 extended abstracts on Human factors in computing systems*, ACM, New York, NY, USA, pp. 491–492.

Reas, C., Fry, B. and Maeda, J.: 2007, *Processing: A Programming Handbook for Visual Designers and Artists*, The MIT Press.

Roma, G. and Xambó, A.: 2008, A tabletop waveform editor for live performance, *in* A. Camurri, S. Serafin and G. Volpe (eds), *NIME'08: Proceedings of 2008 conference on New interfaces for musical expression*, Genova, Italy, pp. 249–252.

Rubin, J.: 1994, *Handbook of usability testing*, John Wiley & sons.

Schafer, M. R.: 1993, *The Soundscape: Our sonic environment and the tuning of the world*, Destiny Books.

Shneiderman, B.: 2007, Creativity support tools: accelerating discovery and innovation, *Commun. ACM* **50**(12), 20–32.

Stowell, D., Plumbley, M. D. and Bryan-Kinns, N.: 2008, Discourse analysis evaluation method for expressive musical interfaces, *in* A. Camurri, S. Serafin and G. Volpe (eds), *NIME'08: Proceedings of 2008 conference on New interfaces for musical expression*, pp. 81–86.

Sutherland, I. E.: 1963, Sketchpad: A man-machine graphical communication system, *in* E. C. Johnson (ed.), *Proceedings of the 1963 Spring Joint Computer Conference*, Vol. 23, American Federation of Information Processing Societies, Spartan Books, pp. 329–346.

Tohidi, M., Buxton, W., Baecker, R. and Sellen, A.: 2006, User sketches: A quick, inexpensive, and effective way to elicit more reflective user feedback, *NordiCHI '06: Proceedings of the 4th Nordic conference on Human-computer interaction*, ACM Press, New York, NY, USA, pp. 105–114.

Ullmer, B. and Ishii, H.: 2000, Emerging frameworks for tangible user interfaces, *IBM Syst. J.* **39**(3-4), 915–931.

Xenakis, I.: 1992, *Formalized Music: Thought and Mathematics in Music, revised edition*, Pendragon Press.

Xu, X., Liu, W., Jin, X. and Sun, Z.: 2002, Sketch-based user interface for creative tasks, *5th Asia Pacific Conference on Computer Human Interaction (APCHI02)*, Beijing, China.

Zbyszyński, M.: 2008, An elementary method for tablet, *in* A. Camurri, S. Serafin and G. Volpe (eds), *NIME'08: Proceedings of 2008 conference on New interfaces for musical expression*, Genova, Italy, pp. 245–248.

Appendix A

Questionnaire

Please, fill in.

A.1 Personal profile

Name: ____________________________

Surname: ___________________________

Age: _______

Do you have any musical training?

◯ No
◯ Less than 2 years
◯ Between 2 and 6 years
◯ More than 6 years

Do you have any experience making music with computers?

◯ No
◯ I've tried some program(s)
◯ I make music with a computer occasionally
◯ I make music with a computer regularly

A.2 Sketching by editing a sound

TASK 1: Load a sound from the library. You can modify it creatively until you are happy with the results (it will be played within a loop). It is possible to erase, copy and paste, draw with the pencil and apply effects.

(Please, think aloud while doing the task).

Interface 1a (*waveTable*). Duration: ______

Interface 1b (*Audacity*). Duration: ______

Final questions Interface 1a-1b

1. Could you find the options without guidance? [LEARNABILITY]
   Interface 1a
   (very difficult) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (very easy)
   Interface 1b
   (very difficult) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (very easy)
2. Is the interface adequate for the task? [USELFUNESS]
   Interface 1a
   (completely disagree) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (completely agree)
   Interface 1b
   (completely disagree) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (completely agree)
3. Does the interface invite you to explore and discover? [USELFUNESS]
   Interface 1a
   (completely disagree) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (completely agree)
   Interface 1b
   (completely disagree) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (completely agree)
4. Could you fulfill your objectives? [USELFUNESS]
   Interface 1a
   (completely disagree) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (completely agree)
   Interface 1b
   (completely disagree) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (completely agree)
5. Do you think the interface is useful? [USELFUNESS]
   Interface 1a
   (completely disagree) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (completely agree)
   Interface 1b
   (completely disagree) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (completely agree)
6. Are you satisfied with the results produced? [EXPERIENCE]
   Interface 1a
   (completely disagree) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (completely agree)
   Interface 1b
   (completely disagree) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (completely agree)

7. Do you like the interface? [EXPERIENCE]
   Interface 1a
   (completely disagree) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (completely agree)
   Interface 1b
   (completely disagree) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (completely agree)
8. Would you incorporate this tool to your (music) studio? [EXPERIENCE]
   Interface 1a
   (completely disagree) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (completely agree)
   Interface 1b
   (completely disagree) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (completely agree)
9. Did you require assistance? Where? (optional)
   ____
   ____
   ____
10. Did you miss any functionality? Which? (optional)
    ____
    ____
    ____
11. Comments (optional)
    ____
    ____
    ____

A.3 Sketching by drawing a sound

TASK 2: Draw sounds on the canvas until you are happy with the composition.

(Please, think aloud while doing the task).

Interface 2a (*wiiMote*). Duration: ______

Interface 2b (*MetaSynth*). Duration: ______

Final questions Interface 2a-2b

1. Could you find the options without guidance? [LEARNABILITY]
   Interface 2a
   (very difficult) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (very easy)
   Interface 2b
   (very difficult) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (very easy)
2. Is the interface adequate for the task? [USELFUNESS]
   Interface 2a
   (completely disagree) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (completely agree)
   Interface 2b
   (completely disagree) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (completely agree)
3. Does the interface invite you to explore and discover? [USELFUNESS]
   Interface 2a
   (completely disagree) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (completely agree)
   Interface 2b
   (completely disagree) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (completely agree)
4. Could you fulfill your objectives? [USELFUNESS]
   Interface 2a
   (completely disagree) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (completely agree)
   Interface 2b
   (completely disagree) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (completely agree)
5. Do you think the interface is useful? [USELFUNESS]
   Interface 2a
   (completely disagree) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (completely agree)
   Interface 2b
   (completely disagree) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (completely agree)
6. Are you satisfied with the results produced? [EXPERIENCE]
   Interface 2a
   (completely disagree) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (completely agree)
   Interface 2b
   (completely disagree) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (completely agree)
7. Do you like the interface? [EXPERIENCE]
   Interface 2a
   (completely disagree) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (completely agree)
   Interface 2b
   (completely disagree) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (completely agree)

8. Would you incorporate this tool to your (music) studio? [EXPERIENCE]

   Interface 2a

   (completely disagree) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (completely agree)

   Interface 2b

   (completely disagree) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (completely agree)

9. Did you require assistance? Where? (optional)

   ____

   ____

   ____

10. Did you miss any functionality? Which? (optional)

   ____

   ____

   ____

11. Comments (optional)

   ____

   ____

   ____

A.4 Sketching by sampling sounds

TASK 3: Record two to four sounds from around (one per track), and mix them until you are happy with the composition.

(Please, think aloud while doing the task).

Interface 3a (*Ableton Live*). Duration: _______

Interface 3b (*tr4ck*). Duration: _______

Final questions Interface 3a-3b

1. Could you find the options without guidance? [LEARNABILITY]

   Interface 3a

   (very difficult) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (very easy)

   Interface 3b

   (very difficult) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (very easy)

2. Is the interface adequate for the task? [USELFUNESS]
   Interface 3a
   (completely disagree) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (completely agree)
   Interface 3b
   (completely disagree) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (completely agree)
3. Does the interface invite you to explore and discover? [USELFUNESS]
   Interface 3a
   (completely disagree) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (completely agree)
   Interface 3b
   (completely disagree) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (completely agree)
4. Could you fulfill your objectives? [USELFUNESS]
   Interface 3a
   (completely disagree) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (completely agree)
   Interface 3b
   (completely disagree) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (completely agree)
5. Do you think the interface is useful? [USELFUNESS]
   Interface 3a
   (completely disagree) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (completely agree)
   Interface 3b
   (completely disagree) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (completely agree)
6. Are you satisfied with the results produced? [EXPERIENCE]
   Interface 3a
   (completely disagree) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (completely agree)
   Interface 3b
   (completely disagree) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (completely agree)
7. Do you like the interface? [EXPERIENCE]
   Interface 3a
   (completely disagree) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (completely agree)
   Interface 3b
   (completely disagree) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (completely agree)
8. Would you incorporate this tool to your (music) studio? [EXPERIENCE]
   Interface 3a
   (completely disagree) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (completely agree)
   Interface 3b
   (completely disagree) ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 (completely agree)

9. Did you require assistance? Where? (optional)

10. Did you miss any functionality? Which? (optional)

11. Comments (optional)