

KEYed User Interface: An HCI Theoretic Design of a Novel

Music Composition Interface

by

Mohamed Ali, Mohamad Farhan

B.E., Bharathiyar University, 1995

A thesis submitted in partial fulfillment of the requirements for
the degree of

Master of Applied Science

in

The Faculty of Graduate Studies

Department of Electrical and Computer Engineering

We accept this thesis as conforming to the required standard:

The University of British Columbia

April 2003

© Mohamed Ali Mohamad Farhan, 2003

In presenting this thesis in partial fulfilment of the requirements for an advanced degree at the University of British Columbia, I agree that the Library shall make it freely available for reference and study. I further agree that permission for extensive copying of this thesis for scholarly purposes may be granted by the head of my department or by his or her representatives. It is understood that copying or publication of this thesis for financial gain shall not be allowed without my written permission.

Department of Electrical & Computer Engineering

The University of British Columbia
Vancouver, Canada

Date April 24, 2003

Abstract

User interfaces in music composition workstations have become cumbersome, especially as they require the use of multiple input devices, such as an electronic piano keyboard, a computer keyboard, and a mouse, repetitively during a composing task. Considering this, our goal is to give the composer a more transparent interface which allows him to focus on the creative aspects of music composition.

Novel and intuitive interfaces for music composition workstations can be designed by applying the findings and principles from the field of human computer interaction. One such design is the **KEYed user interface**, an ergonomic method for controlling music composition software from an electronic piano keyboard by adding a momentary foot pedal as a mode switch. Features for complex sound editing and control are integrated into the system; therefore, the user interface requires far fewer operations to achieve various music production tasks. This helps the composer focus on musical, rather than operational, issues.

The results from our experiments with the **KEYed** user interface show that composers are able to perform production tasks faster when compared to their performance with present user interface setups. Further, they experience enormous comfort, naturalness and intimacy when being engaged with the new interface. The piano keyboard mappings, combined with a single-point touchpad for performing multi-degree of freedom tasks, provide increased speed and intimacy with the controls for improving comfort, thereby enhancing the expressivity of the composer during composition.

Contents

Abstract	ii
Contents	iii
List of Tables	vi
List of Figures	vii
Acknowledgements	ix
1. Introduction	1
1.1 The Need for a New Interface.....	3
1.2 KEYed User Interface.....	4
1.3 Summary.....	7
2. Related Work	8
2.1 Alternative Music Production Input Devices.....	9
2.2 Applying Human Computer Interaction Principles	14
2.2.1 User Centered Interface Design	15
2.2.2 Tacit Knowledge and Metaphors	17
2.2.3 Mode Errors and their Prevention.....	21
2.2.4 Auditory Icons and Earcons as Feedback in Interfaces	24
2.2.5 Space - Time Multiplexing Tradeoffs in Input Devices	28
2.2.6 Human Manual Studies.....	30
2.3 Summary	36

3.	KEYed User Interface	37
3.1	KEYed User Interface Prototype I.....	37
3.2	KEYed User Interface Prototype II.....	38
3.3	KEYed User Interface System.....	39
3.3.1	Hardware Layout	39
3.3.2	Piano Keyboard Mapping Design.....	41
3.3.3	Software Design.....	52
4.	KEYed User Interface Experiments	55
4.1	Preliminary Studies with Prototype I.....	55
4.1.1	Goals and Hypothesis of the Study.....	55
4.1.2	Participants.....	56
4.1.3	Controlled Experiment Design	58
4.1.4	Analysis of Results	59
4.2	Final Studies with Prototype II	60
4.2.1	Goals and Hypothesis of the Studies	61
4.2.2	Participants.....	61
4.2.3	Controlled Experiment Design	64
4.2.4	Analysis of Results	69
4.2.5	Case Study with Prototype II.....	70
4.2.6	Case Study Results.....	72
4.3	Comparison of the Predicted and Measured task Completion Times.....	75
5.	Discussions	75
5.1	Preliminary Studies with Prototype I.....	75
5.2	Controlled Experiments with Prototype II.....	76

5.3	Case Study with Prototype II	81
5.4	Comparison of the Predicted and Measured task Completion Times.....	82
6.	Conclusion and Future Work	84
6.1	Overview and Conclusion.....	84
6.2	Future Work	86
6.3	Contributions.....	87
	Bibliography	88
	Appendix A KEYed User Interface Experiments	94
A.1	Participant Questionnaire for Prototype I	94
A.2	Task A used for Prototype I.....	96
A.3	Task B used for Prototype I	98
A.4	Questionnaire A for Prortotype II.....	100
A.5	Questionnaire B for Prototype II.....	106
A.6	Task A used for Prototype II.....	113
A.7	Task B used for Prototype II.....	114
A.8	Task C used for Prototype II.....	115
A.9	Consent form for the Prototype II Controlled Experiment	117
A.10	Consent form for the Prototype II Case Study	120
A.11	Comments on the Questionnaires	123

List of Tables

2.1	Estimated Times for the K, B, P, H and M Operators	35
3.1	Window - Octave Layout.....	42
3.2	Mixer / EQ Octave Layout.....	44
3.3	Transport Octave Layout	46
3.4	Arrange Octave Layout.....	47
3.5	Key Edit Octave Layout.....	49
3.6	Edit / File Octave	51
4.1	ANOVA Summary for all the Interactions	67
4.2	ANOVA Summary for Keyboard / Mouse Interaction vs <i>KEYed</i> (Audio).....	68
4.3	ANOVA Summary for Keyboard / Mouse Interaction vs <i>KEYed</i> (No Audio)....	68
4.12	Measured Mean Task Completion Time vs Predicted Keystroke-Level Model (KLM) Task Completion Time (A, B).....	73

List of Figures

1.1	Current Digital Music Workstation Setup	3
1.2	KEYed User Interface Layout	5
1.3	KEYed User Interface Setup.....	6
2.1	Human User Interface by Mackie Designs®	9
2.2	Pro Control® Surface by Digidesign®	10
2.3	Space Multiplexed Mixer (a), Time Multiplexed Mouse (b).....	29
3.1	KEYed User Interface Prototype I.....	38
3.2	KEYed User Interface Prototype II.....	38
3.3	Single Point Touchpad.....	39
3.4	Hardware Layout for Prototype I.....	40
3.5	Hardware Layout for Prototype II.....	41
3.6	KEYed User Interface Piano Mappings.....	42
3.7	Mixer / EQ Octave Layout.....	44
3.8	Editing with Miser / EQ Octave.....	45
3.9	Transport Octave Layout	46
3.10	Transport Window	47
3.11	Arrange Octave Layout.....	48
3.12	Arrange Window.....	48
3.13	Key Edit Octave	49
3.14	Key Edit Window	50
3.15	Drawing Volume Graph.....	50
3.16	Edit / File Octave Layout	51
3.17	KEYed User Interface Software Design	53
4.1	Mean Reaction Time with 95% Confidence Error Bars	58

4.2	Mean Subjective Rating	59
4.3	Prototype II Experiment Setup.....	63
4.4	Prototype II Piano and Pedal Setup	63
4.5	Mean Completion Times of the Task Complexities for the Different Interaction Methods for 10 Composers.....	64
4.6	Mean Completion Times of the Interaction Methods for the Different Task Complexities for 10 Composers.....	65
4.7	Interaction Graph A to Visualize Interaction Effects between the Task Complexities and the Interaction Methods	66
4.8	Interaction Graph B to Visualize Interaction Effects between the Task Complexities and the Interaction Methods	66
4.9	Task Completion Times for the Different Interaction Methods for the Simple Task, Before and After the Case Study for One Composer.....	71
4.10	Task Completion Times for the Different Interaction Methods for the Moderate Task, Before and After the Case Study for One Composer.....	71
4.11	Task Completion Times for the Different Interaction Methods for the Complex Task, Before and After the Case Study for One Composer.....	72

Acknowledgements

I would like to express my sincere gratitude to Dr. Sidney Fels for giving me a chance to work under him, and for providing me with excellent guidance, encouragement, and support during the last two years. I could not have asked for a better supervisor.

I would also like to acknowledge the support and help which I got from Dr. Brian Fisher and Dr. Karon MacLean. A special thanks to Graeme McCaig for his equipment support, and Edgar Flores and Tim Chen for technical support in the HCT lab. I also want to acknowledge ATR, Japan and NSERC for providing financial support and equipment for this research. I would also like to thank all the members in the HCT lab for their valuable comments and suggestions, and my friends for their support.

Finally, though I cannot express it in words, I express my heart felt gratitude and indebtedness to my parents, brothers and to my girlfriend, who have all encouraged me and helped me in my endeavors.

FARHAN MOHAMED

Chapter 1

Introduction

Let us start with a scenario from the Project KK music production studio:



Composer and Producer Kevin, who has been hired to compose the music track for the new *POTS curry flavor potato chip* commercial, is excited about this brilliant musical score emerging in his mind. Before this burst of creative musical ideas fades away, he wants to quickly record and produce these with his music composition workstation.

- *He starts by opening a new project page and a new track on the Cubase® music software, and clicks the record button on the transport window to capture his performance. He then begins to perform his ideas with an electronic piano keyboard, which triggers a piano sound module connected to the workstation software.*
- *When he realizes that he needed to use a particular EQ (equalizer) setting on that track to enhance the sound of the piano, he stops playing, opens the EQ window on*

the Cubase® software using a macro on the computer keyboard, moves the mouse cursor to the EQ knob and holds down the left mouse button to turn the knob to achieve the required sounds. He then returns to the piano and lays down a beautiful melody.

- *Though he is very satisfied with the recorded piece of music, he wants to edit two notes by extending their lengths. He opens the Key Edit window on the Cubase® software with a computer keyboard macro, holds the mouse cursor on top of the notes and stretches these by holding down the left mouse button. He then continues to play until further editing is required to achieve his original musical idea.*

We see in this scenario that Kevin uses different computer software and hardware tools repetitively to aid his creative work. He uses a music software system for recording and editing sonic materials, patching effects and dynamic processor plug-ins, panning sounds, automating mixers, and so on. To interact with the software system, he uses three input devices, namely, an electronic piano keyboard for entering music related data (notes or performance data), a computer standard QWERTY keyboard, and a mouse for editing and processing such data.

The use of such computer tools is also due to the modern trend of composing or creating musical ideas with various music production functions, such as the equalization and the spatial positioning of the sounds in mind. These production functions have become an integral part of the ideas and constructs of the composers' mind, which are externalized during composing, facilitated by the computer tools. For example, the spatial positioning of a sound, or the use of a certain effect for a sound, is not just something that is added post-synthesis, but plays an equal role along with other parameters, such as fundamental

frequency, formant frequency, and so forth, during music composition [5]. Taking this into account, along with the repetition involved in performing such production functions as shown in our scenario, we contend that the task of performing, such as playing a piano, and the task of producing or editing, such as spatial positioning of the sound are bound together and performed in real time, as part of the composing process.

In essence, we see that the overall task of composing with music software requires several repetitions of subtasks before a satisfactory musical product is achieved. An example of a subtask is using a computer keyboard macro to open the EQ window to adjust the frequency settings of the sound as shown in our scenario.

1.1 The Need for a New Interface

Due to such repetitive practices, existing user interfaces in music workstations have become cumbersome, as they require switching between multiple input devices, such as an electronic piano keyboard, a computer keyboard, and a mouse, as shown in Figure 1.1. (In music workstations, the primary instrument used to input musical notes or performance data is often referred to as a master controller; in our case it is an electronic piano keyboard).

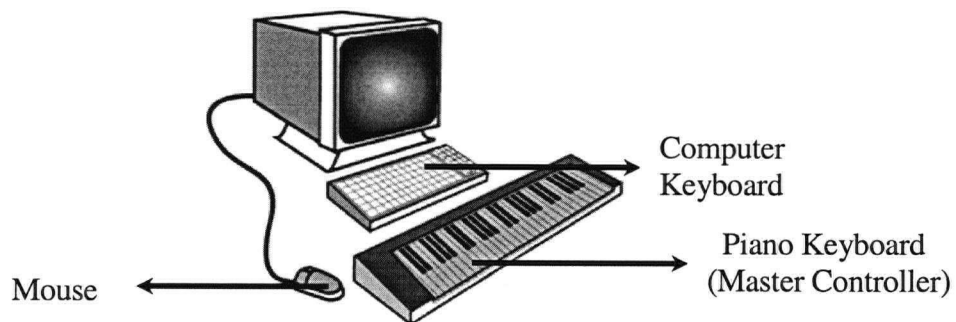


Figure 1.1: Current Music Workstation Setup

The disadvantage of having more input devices is the excessive physical space they occupy and the discomfort in using them due to their placements, especially in the current music workstation setup. More importantly, the composers' creative work is constantly interrupted by the time spent switching between multiple input devices and figuring out their functions.

1.2 KEYed User Interface

Taking these facts together, we realize that new input devices for performing such music production functions may greatly improve the ease of use of production workstations, and enhance the overall musical expression of the composer. We observe that it is possible to reduce the number of input devices used by moving some of the more commonly used computer keyboard macros and the mouse functions to the master controller. This allows the composer to work more efficiently, because all common functions can then be accessed using only the master controller.

This observation gave birth to the *KEYed* user interface. The *KEYed* user interface provides a mapping of the music production functions from the piano keyboard itself. This is done using an octave structure, with a key based segmentation. The customizable piano mapping provides the composer with a familiar configuration of space and sound, allowing him or her to focus on the creative aspects of music composition. We anticipate that the user interface would require far fewer operations to achieve various production tasks, thereby helping the composers focus on musical rather than operational issues.

For example, a macro that can be relocated to the piano keyboard is the copy function, or [Control]-[C], which copies a selected sequence to the virtual clipboard. Such functions are relocated to specific octaves on the piano keyboard. The default mappings of

the functions to the octaves of the controller are detailed in Chapter 3. To distinguish between the piano keystrokes that represent notes, and those that represent macros, a momentary foot switch is used as a mode switch, thereby reducing mode errors [1]. Though a secondary body channel, such as the foot, has a lower information processing bandwidth, [25] it sets the framework and the reference [22] for the primary body channel, such as the hands as shown in Figure 1.2. In addition, the system state is reinforced by the foot proprioceptive feedback [1], and the relocation of the functions to a single input device could reduce the device acquisition time [2]. Even when the foot switch is depressed, the piano key strokes produce audible notes. This may act as a mnemonic aid (by the creation of earcons [6]) to the composer as to which functions he is ‘performing’. This is one aspect we investigated as discussed in Chapter 5, Section 5.2.

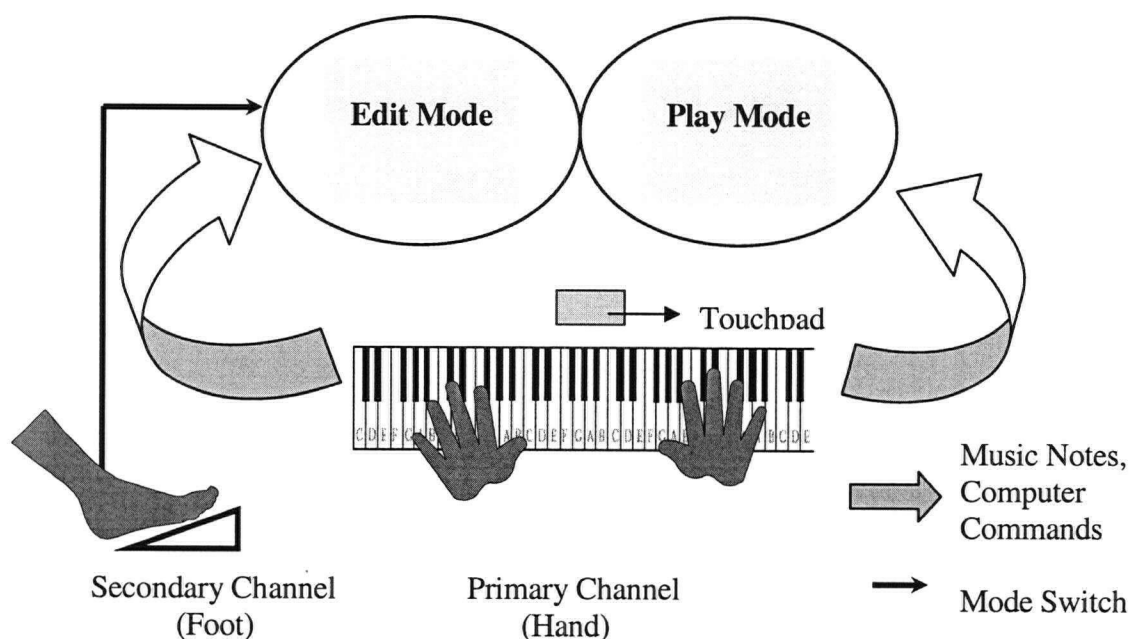


Figure 1.2: KEYed User Interface Layout

Further, the KEYed user interface has a single-point touchpad placed onto the piano keyboard, as shown in Figure 1.3. The single-point-touchpad is integrated for the fine

control of selected continuous parameters. The touchpad is used for constrained vertical or horizontal actions, for a single degree of freedom (DOF) task, such as sliding a fader, and to perform a full 2DOF task, such as drawing a modulation graph after the required parameters for control, are selected with the piano notes. While the touchpad is a new device for the composer, we meticulously place it to minimize the device acquisition time and is investigated in our studies for its effectiveness as discussed in Chapter 5, Section 5.2.

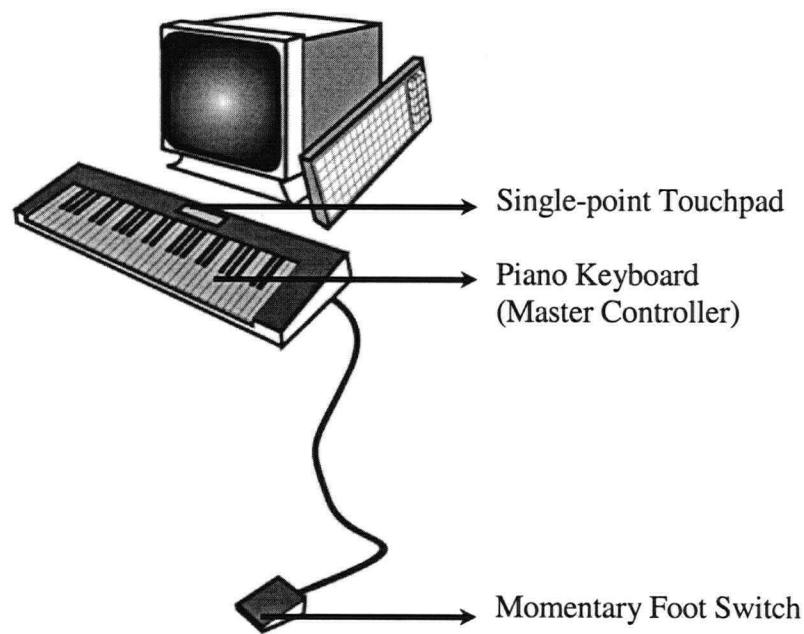


Figure 1.3: KEYed User Interface Setup

As part of our requirements analysis, we discussed the trends and practices in music composition with music producers and composers at the Trebas Institute, a premiere recording institute in Vancouver. Careful observations were done on the composers' current music workstation setups and frequently used production functions.

1.3 Summary

In summary, our motivation comes from observing the awkwardness of the existing user interfaces in music composition workstations. The goal of our work is to give the composer a more transparent [16] interface, such as the *KEYed* user interface, which allows him to focus on the creative aspects of music composition. In addition, it is hoped that a user oriented interface will minimize muscle stress and reduce any incidence of injuries, such as carpal tunnel syndrome.

In the following chapter, we begin by discussing the reasons why there is minimal research for improving the interfaces used in music composition workstations. The field of Human Computer Interaction (HCI), and its importance and application in composition interface designs are discussed. Further, we give the literature and design tools that we apply from HCI, which are relevant and useful in the design of the *KEYed* user interface. Results from classical HCI are used as tools for developing methodologies for the design and evaluation of the *KEYed* user interface. We also furnish alternative computer interface designs, introduced by music production software manufacturers and HCI researchers that are relevant to our work in Section 2.1.

Chapter 2

Related Work

Human computer interfaces, or in our case, composer computer interfaces, used in music production workstations, take its current shape from the existing user interface research applied to common computing environments. The use of the computer keyboard and mouse as physical handles to music sequencing and production programs have existed since the use of software programs, to arrange and edit music.

On the contrary, significant user interface research exists in the field of music synthesis and performance. In this field, new interfaces for musical expression are typically thought about in the context of real-time performances. Interface designers for the live performance of computer music borrow tools and principles extensively from HCI, for dealing with such specific topics as simultaneous multi-parametric control, timing and rhythm. Various input devices and controllers are designed and developed for specific artistic demands. As a result, various principles for designing computer music controllers are proposed by expert researchers [35].

However, little research has been done in improving composer computer interfaces due to the notion that composition software program is more of an off-line music editing tool than its synthesis and performance counterpart, where enormous possibilities exist in the real time control of music synthesis engines and parameter mapping. We contend that the production functions in the modern composition process also provide opportunities for new ways of expression. Functions such as the spatial positioning, fades and mutes of a sound source are performed in real-time during composition as mentioned in Chapter 1.

Further, while performing musical score editing functions during composition, composers are more engaged, with the composition software for tasks such as note searching and score positioning, than music content creation through performance. Thereby, we look at the performance and editing tasks as a single task done in real-time during the overall task of music composition.

Considering these possibilities, we discuss in detail, about the importance and how the principles in the field of Human Computer Interaction (HCI) can be applied to the design of composer computer interfaces in Section 2.2. To begin with, we discuss some interesting composer computer interface designs, recently introduced by music system manufacturers and researchers in the following section.

2.1 Alternative Music Production Input Devices

Efforts to improve composer computer interaction have been addressed recently in the music software industry. One such example is the Mackie's Human User Interface® (HUI), as shown in Figure 2.1.



Figure 2.1: Human User Interface by Mackie Designs®

The Human User Interface relocates the most commonly used music production functions, if not entirely, onto its control surface. The control surface design replicates the music production software in its design with channel layouts, transport controls, faders for mixing, and dials and buttons for selection and scrolling. There is a one to one mapping of the controls on the software and the controls on the human user interface hardware. The primary limitation of this interface, apart from the cost, is that the design takes up additional space in the composers' studio and creates a split in the performance and editing tasks during music composition. Further, with the increasing complexity in music composition software, time multiplexing of function controls, as we discuss in the Sub-section 2.2.5, may be required in the interface.

Such designs are borrowed from the recording industry. With digital recording consoles getting increasingly popular, more research has been done on improving the user interface for such systems by facilitating a console to mix and route various musical tracks, which is its primary function, and for controlling the software functions to record and produce the tracks on a digital audio workstation. One such product is the Pro Control® hardware surface for Digidesign's Pro Tools® workstation, as shown in Figure 2.2.



Figure 2.2: Pro Control® Hardware Surface by Digidesign®

These recording consoles facilitate the complete control of the digital audio workstation software from the recording console, which frees the user from having to use a computer keyboard and mouse. The design of the *KEYed* user interface is similar in its design principle in that it gives a composer software function controls from the piano keyboard, which is her primary work surface, and is analogous to the audio engineer using the Pro Control® to record, edit and mix audio tracks.

Further, multi-parameter controllers for recording consoles using a higher degree of freedom controllers (gesture based input devices) combined with multiple feedback mechanisms (visual and haptic feedback) for sound spatialization tasks are also investigated by human computer interaction researchers [37].

Other alternative interface design approaches in music composition workstations integrates the performance and editing tasks during music composition. Such a design is adopted by the E-Magic® software, called Logic Audio®, a popular music production system that allows the composer to map the most commonly used computer keyboard macros used during editing to the higher octaves of the piano keyboard. This design assumes that the composer does not use the mapped octaves while recording with the music workstation, and as a result, it creates the following:

1. A cognitive split while composing as the composer needs to be aware of the higher octaves.
2. Dramatic mode errors while recording.

On the contrary, the *KEYed* user interface, as we discussed in Chapter 1, has functions mapped to all the octaves of the piano keyboard facilitated by the momentary foot switch, thereby minimizing such mode errors.

At the Drake Music Project (DMP), a charity established to facilitate music making for physically disabled people through the use of music technologies, a variety of hardware and software are designed to facilitate or assist in the mouse and computer keyboard usage of music systems [4]. Switch controlled menu-driven overlays to emulate key press actions, external switches to replace mouse buttons, extensive use of macros and joysticks, or switches to emulate mouse movements are some of the adaptations used. Interestingly, while such adaptations are found to be useful for the disabled, they are also found to be too restrictive when used with a WIMP (Windows, Icons, Menus, Pointers) based music system, such as the modern music workstations, mainly due to the vast choice of actions presented to the composer.

More recently, research on specific functions, such as digital audio navigation in music workstations, has been done using haptic (sense of touch) technologies. Traditionally, with acoustic music instruments and magnetic reel editing, performers and music editors have had an intimate relationship with their instruments and magnetic reels. They rely not only on visual and aural feedback, but also on haptic feedback while performing or editing. However, with modern music workstations, this close physical relationship disappears since the haptic feedback felt through a keyboard or mouse is not representative of a sound file or MIDI instructions. A haptic knob is used to facilitate audio navigation using various displays, such as pops, detents and textures, and is currently being developed and evaluated for its effectiveness in the music editing domain [36].

An alternative approach to the current user interface design in music editing is the use of voice input for communicating with the composition software. Speech interfaces are often characterized as being 'more natural' than other types of input devices like the

keyboard, mouse and touch-screen. Studies [45] have demonstrated the improved efficiency of speech over other modalities for human-human communication. The underlying assumption for the desirability of human-computer speech interfaces is that the skills and expectations that users have developed through everyday communication will result in speech input being more efficient and effective than alternative methods. Further, speech interfaces in computer workstations facilitate direct access to virtual functions and operations, as opposed to graphical user interfaces with windows, icons, menus and pointers (WIMP) which require physical acquisitions of input devices and graphical navigation through the WIMP interface to access the functions. However, one serious limitation of speech interfaces is the lack of adequate feedback mechanisms to avoid errors during interaction. For example, when a composer says or inputs the phrase '*Lets draw a volume graph for track 1*', mode errors can occur if the composer is not aware that he is in the volume graph edit mode while he uses the mouse or a touchpad for other tasks. Such effects can be avoided by using sustained kinesthetic feedback mechanisms during the entire edit mode transaction. The KEYed user interface adopts such a feedback mechanism as the foot is engaged with the momentary foot pedal as discussed in Chapter 1, throughout the edit mode. The other disadvantage of using speech input is due to the interference from other musical and non-musical sounds present during music performance and playback. One way to avoid this is to use close microphones and headphones while using a speech interface, which we contend is not appropriate during a composition task. However, considering the effectiveness of speech over other modalities as discussed earlier, we anticipate that the speech input can complement the KEYed user interaction due to the

sustained feedback facilitated by the interface. Appropriate use of speech input while editing using the *KEYed* user interface need to be explored in the future.

2.2 Applying Human Computer Interaction Principles

Although some of these alternative input device designs do emerge from research in the field of Human Computer Interaction (HCI), we see various limitations in them which we believe is due to the lack of applying the principles in HCI adequately and appropriately. We believe that significant improvements can be achieved by giving careful attention to the user and her acquired subjective knowledge, and by providing effective feedback mechanisms, which are some of the issues well researched by HCI practitioners. Results from such research may be used as tools for developing methodologies for the design and evaluation of composer computer interfaces, considering important factors, such as the main channel of communication (visual, auditory, haptic), the goal of interaction (a work to be done or artistic expression), and users and their expected level of expertise.

In essence, a substantial amount of material has been published in the HCI literature on the evaluation of existing input devices, as well as on the design of new ones. The design of input devices used in music production workstations can be benefited by carefully applying such findings. The following sub-sections detail the literature on some of the design principles, tools and findings, we apply from HCI research that is relevant and useful in the design of the *KEYed* User Interface. We begin with one of the key design principles in HCI, which explains the importance and processes involved in the early focus on users.

2.2.1 User Centered Interface Design

The largest step towards a user centered human computer interface design is the field of Human Computer Interaction (HCI). The first principle of any user centered design principle in HCI literature has to do with the users themselves. The objective here is to understand the user's cognition, behavior, tacit knowledge and emotion. Norman [10] refers to the science of interface design as 'cognitive engineering.' This requires, he states, some formal models of people and of interaction, models that need not only be approximations, but that are precise enough to lead to design rules.

Norman's principles are conceptual in nature as they reside on a fairly high level. We want to pay attention to the lower level, practical principles for involving composers in the interface design. Preece [11] mentions the following important design principles:

1. Early design is dominated by collecting and synthesizing information about users' needs and capabilities [11]. The main techniques for obtaining this information are the following:

- a) **Requirements analysis:** In this stage, information is typically acquired by using interviews and questionnaires or by observing and analyzing current practice. This is our first step in the design of the *KEYed* user interface, as mentioned in Chapter 1. The primary focus of our requirements analysis was to identify the most commonly used music workstation functions while composing. For example, some of the commonly used tasks we identified while composing are

1. Performing a recording for a preset length of time and playing back the recorded section.
2. Adjusting the volume fader.

3. Editing notes for its length and so on.

Further, careful observations are made on the composer's current music workstation setup and work practices, followed by informal interviews.

b) **Task analysis:** In this analysis, details of the user's task and information about the task environment are collected, so that the user's needs are well understood. Preece distinguishes macro methods, in which the whole system is analyzed in terms of organizational, social, and environmental aspects, from micro methods in which discrete tasks are decomposed into hierarchical structures and then finally into small cognitive units. Examples of the latter include the well known GOMS family of models (Goals, Operators, Methods, and selection Rules) [38]. A more appropriate model we use for predicting task completion times with the *KEYed* user interface is the Keystroke Level Model (KLM) [39]. KLM is a simplified version of GOMS [38] in that it focuses on very low level tasks. The actions are termed keystroke level if they are at the level of actions similar to pressing keys, moving the mouse, pressing buttons, and so forth as discussed in the Sub-section 2.2.6 of this chapter. The predicted KLM task completion times for the various tasks used in our studies are further discussed in Chapter 4.

c) **Usability tests:** This technique is often applied for evaluation purposes to provide information for the upgrading and maintenance of existing systems. Most usability testing involves experimental tasks that reflect important and frequent uses of the system. When usability testing is integrated into the design cycle by being quantitatively specified in advance, it is known as usability engineering [12]. Usability tests performed on the different prototypes of the *KEYed* user interface

are detailed in Chapter 4. A preliminary usability test and evaluation of the initial prototype of our interface identified the ways for improving the final prototype as discussed in Chapter 5.

2. As the design process develops, the design will be transformed through various forms of specification (e.g., a natural language description or all sorts of diagrams, or even some form of non-executable prototype that is presented with video equipment) and prototypes [11].

In essence, we have seen the importance and usefulness of a user centered design and how the design principles suggested by Preece [11] is used in our design of the *KEYed* user interface. In the following sub-sections, we discuss in detail, some of the key issues and findings in the HCI literature that are relevant and useful in the design of the *KEYed* user interface.

2.2.2 Tacit Knowledge and Metaphors

Focusing more on the users, it is important to look at a type of knowledge that is tacit in composers, which can be beneficial while designing composer computer interfaces. A term which is introduced by Polanyi [42], ‘tacit knowledge’ is defined as the knowledge that enters into the production of behaviors and/or the constitution of mental states but is not ordinarily accessible to consciousness. It is often described as subjective knowledge as it is composed of insights, intuitions, and emotions, which are some of the common characteristics found in musicians and other artists.

To give an overview, there are the following four types of user knowledge proposed by knowledge researchers [33], which we should be aware of as user interface designers:

1. The user knows that he knows - explicit knowledge.

2. The user knows that he does not know - explicit knowledge of ignorance.
3. The user does not know that he knows - operational knowledge.
4. The user does not know that he does not know - unconscious knowledge.

The 'operational knowledge' is the essential domain of tacit knowledge as it is the area of knowledge by which a performer performs without the awareness of how he or she performs. Researchers call this 'the icebergs of knowledge' representing explicit knowledge, as what is found above the water and the rest being tacit. The distinction between tacit knowledge and explicit knowledge has sometimes been expressed in terms of knowing-how and knowing-that [43], or in terms of a corresponding distinction between embodied knowledge [42] and theoretical knowledge. On this account knowing-how or embodied knowledge is characteristic of the expert, who acts, makes judgments and so forth without explicitly reflecting on the principles or rules involved. The expert works without having a theory of his or her work; he or she just performs skillfully without deliberation or focused attention.

Studies with expert dance performers have shown that the key to the successful acquisition of a dancer's intuition has little to do with past athletic experience, and everything to do with years of practice dedicated to repetition and the analysis of dance phrases [15]. It is this self-motivated discipline and analysis that helps a dancer acquire the tacit physical knowledge required to move from conscious reflection of individual movement to unconscious automatized movement, leaving the mind free to focus on the emotional and expressive aspects of the dance.

Similarly, years of training by musicians provide their mind and body with the tacit muscle memory required to perform automated music structures and units. User interface

designers, especially in a field such as music performance and composition, should develop techniques for dealing with such knowledge and its externalization. For example, during music performance, musical structures and units are retrieved from memory according to the performers' conceptual interpretations, and are transformed into appropriate movements [15]. The performers' motor program and muscle memory contains representations of an intended action and processes these into a movement sequence. We see how such acquired tacit muscle memory helps the composer perform editing tasks faster with the *KEYed* user interface in Chapter 5.

System designers deal with tacit knowledge by trying hard to make knowledge explicit; however, if possible, it should be left as tacit, and if not possible, then the system should be built around practices that aid in the discovery and expression of tacit knowledge [33]. Four modes of knowledge conversions are summarized in the case study done at the Knowledge Creating Company [32]. They define the process of articulating tacit knowledge into explicit concepts as 'externalization'. Such externalization often involves the creation of a 'metaphor' or an analogy. Such metaphors are an important kind of learning, used quite often in teaching. The notion of employing metaphors for interface design has partially replaced the notion of the computer as the tool with the idea of the computer as a presenter of a virtual world or system, in which a person may interact more or less directly with the representation. As Weiser [14] explains, a good tool is an invisible tool. As an example of a good tool, he mentions eyeglasses: you look at the world, not at the eyeglasses.

Further, Carroll et. al. [13] suggest the following four steps for designing interfaces in which metaphors are used:

1. The identification of candidate metaphors.
2. The detailing of the metaphor/software matches with respect to representative user scenarios.
3. The identification of inevitable mismatches and their implications.
4. The identification of design strategies to help users manage mismatches.

According to Carroll, if only the interface presents representations of real-world objects, people will naturally know what to do with them [13]. An ideal example is the piano keyboard metaphor used in the *KEYed* user interface as discussed in the Chapter 4. We use the piano keyboard metaphor in the user interface to manipulate software objects. Our expectation is that the composers' acquired spatial and auditory knowledge of the piano keyboard would facilitate a natural interface to perform such operations and further provide ways to externalize their tacit knowledge.

Further, the design elements suggested by Heckel [31] reinforce the importance of these issues. An example of design elements includes the following:

1. Leverage the user's knowledge
2. Speaking the user's language
3. Communicate with metaphors

Designs considering such an acquired subjective knowledge of the users, their work contexts and the use of metaphors, give rise to intimate relationships between the user interface and the body channels engaged with it. Further, aesthetics flow from such intimacy [16].

The following two sub-sections address the importance of appropriate feedback mechanisms in user interfaces for eliminating operational errors in music composition workstations.

2.2.3 Mode Errors and their Prevention

In designing user interfaces it is important to be aware of a large class of errors that occur due to modes. Errors occur while performing a certain operation for one mode when in fact, you are in another. Norman (1981) defined mode errors as the misclassification of a situation resulting in actions that are inappropriate for the true situation [17]. Whenever a particular action has different consequences depending upon the state of the system, mode errors may occur. The classic example of this is in text editors with command-line interfaces. Inserting text in the *vi* editor, a UNIX-based text editing system [40], while in the command mode is a very common mode error we experience. Monk [19] refers to such mode errors as mode ambiguity. His user centered account of mode errors is concerned with the intentions of and the actions taken by a user, as opposed to most definitions he refers to as machine-centered.

Norman [17] suggests that avoiding modes entirely is not practical, but one can minimize modes. The best way to reduce these is to provide continuous and meaningful feedback to the user. Monk [19] suggests reducing mode ambiguities by increasing the 'width' of the user interface, for example, by introducing new keys or additional input devices as alternatives to the keyboard. However, using an unfamiliar device imposes additional cognitive demands, which slows the performance on all aspects of the task, and also outweighs the benefits in reducing mode ambiguities. This is also the limitation of the alternative interfaces discussed in the Section 2.1. Further, widening the user interface will

always has its costs as well as benefits, and their extent very much depend on the particular application. One solution to this problem is to design systems considering factors such as the users' familiarity with other systems. Such a design is considered in our design of the *KEYed* user interface.

In the analysis of mode errors, it is found that the mode ambiguity only leads to mode errors when it results in user expectations, which do not correspond to the actual system effect. To achieve a general awareness of such mode changes, they must be signaled to the user as clearly as possible. Results of one experiment [19] using sound to signal mode changes, shows that sounds can be effective for signaling modes. The number of mode errors made is dramatically reduced by using such audio feedback. We have explored such auditory feedback mechanisms for its effectiveness in the *KEYed* user interface as discussed in Chapter 4.

In other studies, pressure and movement feedback has been shown to be effective in reducing mode errors in common text editing tasks [1]. The effectiveness of kinesthetic versus visual feedback is compared over different conditions involving the use of the keyboard (with insert and command mode keys) versus a momentary foot pedal (like a piano sustain pedal) for changing modes (kinesthetic feedback), crossed with the presence or absence of visual feedback to indicate the mode. While both visual and kinesthetic feedback are found to reduce mode errors both in novice and expert users, the results made a stronger case for kinesthetic over visual feedback for the prevention of mode errors. Further, the use of a foot pedal, which is constantly engaged with the user, led to a significantly faster resume time [1] than the computer keyboard. The presence of visual feedback made no difference, and the pedal feedback effectively reduced the cognitive load

imposed by the system. This is because, in the case of the foot pedal, the tool for articulating mode switching is also the limb through which sensory feedback on the mode status is received.

The momentary foot pedal also acts like glue that ties the subtasks, for example, consider the following actions in a sequence:

1. Depress the foot pedal.
2. Open a window in the music production software by holding down a piano note while holding down the pedal.
3. Click a button within the window by holding down another piano note while holding down the first piano note and the pedal.
4. Release the foot pedal.

Similar studies with pop-up menus illustrate this point. For example, though we consider making a selection from a pop-up menu as being a single task, on closer examination, it consists of three subtasks, such as invoking the menu by depressing the mouse button, navigating the selection by moving the mouse while the button is depressed, and making a selection and returning by releasing the mouse button. Interestingly, the tension of holding the mouse button down throughout the transaction is found to be the glue that ties the three subtasks [20]. By designing dialogues in user interfaces in this manner, mode errors and errors of syntax are found to be virtually impossible to make since the concluding action, which is the mouse button release in this case, is the unique and natural consequence to the initial action, which is depressing the mouse button. In our earlier example, the tension on the foot while holding down the momentary foot pedal gives constant feedback illustrating a temporary state or mode, and the release of the pedal completes the transaction.

Such a detailed study of mode errors and the appropriate feedback mechanisms for reducing these, are critical in the design of a multi-modal system, such as the *KEYed* user interface. The *KEYed* system has a hierarchy of quasi-modes [44] consisting of a macro mode and multiple micro modes. The macro quasi-mode exists in the use of the piano keyboard for both the performance task and the editing task, temporarily during composition. While in the edit mode, several quasi-modes exist in the use of specific octaves to perform composition functions, the use of a certain note to open a specific composition software window and the use of other notes to perform edit functions within the window. To distinguish between the piano keystrokes that represent notes, and those that represent macros, a momentary foot switch is used as a mode switch in our design, thereby reducing macro mode errors [1]. Further, the system state is reinforced by the foot proprioceptive feedback, as discussed in Chapter 1 and the earlier part of this sub-section. Similarly the kinesthetic feedback in the finger while holding a note reduces micro mode errors as validated in our experiment results and discussed in Chapter 5.

The following sub-section on the study of the effectiveness of auditory icons and earcons [6] also emerges from the importance of reducing such mode errors. We begin by furnishing a broad outlook of the importance and research findings in the use of auditory feedback to user interfaces, and further, its relevance to our design of the *KEYed* user interface.

2.2.4 Auditory Icons and Earcons as Feedback in Interfaces

The acoustic scenery of our everyday world has a very intricate structure because it is a direct consequence of the complex happenings around us. Listeners are able to 'read' the sound directly and to hear in it, the events that gave rise to it. The stimulus for exploring

these issues has come from the arrival of new technologies, which offer the possibility of combining visually presented information with sound. The need to design better computer systems to extend the range of human thought has encouraged people who understand this acoustic technology to provide more and better information to the user.

The use of non-speech audio at the graphical user interface is becoming increasingly popular due to the potential benefits it offers [7]. There are many reasons for this. In everyday life, people communicate using all their senses, with information in one sensory modality being backed-up by data from the others. When they come to use computers, the interaction gets restricted almost solely to the visual channel, and this limitation can cause the interface to intrude into the task that the user is trying to perform. The aim of a multimedia interface is to make the interaction more natural, and the interface more transparent by using different forms of input and output. Most current interfaces make little use of sounds other than beeps to indicate errors.

Non-verbal sounds, which are sounds other than speech, can be categorized into two broad categories, namely, Auditory icons and Earcons [6]. While Auditory icons can be a broad category of sounds involving synthetic, or naturally occurring sounds (environment sounds), Earcons are abstract, synthetic tones that can be used in structured combinations to create sound messages to represent parts of an interface.

Some of the early research on this was done by Gaver and Smith, who came up with some classic ways to use environmental sounds to enhance the usability of systems which employ multiprocessing and modes, extended or layered displays, and collaborative workspaces [7]. They conducted experiments on a shared virtual environment called SharedArk, which is a collaborative virtual physics laboratory for distance education,

allowing one or a number of users to interact simultaneously. The technique addresses the issues of confirming user-initiated actions, providing information about ongoing processes or system states, providing adequate navigational information, and signaling the existence and activity of other users who they collaborate with in their tasks, but are not visible. Confirmatory sounds used in the interface are designed as analogous to its real world. Using simple everyday sounds, such as taps and clicks, to supplement or replace visual highlighting provides the kind of auditory confirmation that we rely on in the everyday world. Auditory icons and earcons can also serve to remind users' of their continuous states or ongoing processing without depending on windows that are likely to use unnecessary space. This relies on the fact that we quickly notice when there is a change in a continuous sound.

Earcons are also studied for their effectiveness in computer interfaces [8]. Earcons are composed of motives, which are short, rhythmic sequences of pitches with variable intensity, timbre and register. Earcons provide a powerful method of 'sonification'. They can be used for adding sound to both data and interfaces. Related items can be given related sounds, and hierarchies of information can be represented. Complex messages of subunits can be built up. These are powerful and flexible means of creating auditory messages. Experiments show that high levels of recognition can be achieved by the careful use of pitch, timbre and rhythm [8].

Further, interesting studies on the effectiveness of auditory icons have been done by evaluating three different audio environments in a 3D task undertaken by visually impaired people [9]. This is an increasingly important area of study, as user interfaces are becoming more 'graphical' and less text based, and thus hindering the visually impaired user. It is

interesting to see how non-speech audio can enhance the user's perception of depth especially for the visually impaired. Although the time taken for the visually impaired users to locate a position is significantly longer than a sighted user with a display, their accuracy is better than the sighted user. Sighted users relied more on their sight and interestingly, we see that the visually impaired users outperformed them in the accuracy studies. Sighted readers also preferred the musical environment, and the visually impaired preferred the tonal environment [9].

Such studies on the effectiveness of auditory icons and earcons are useful in the design of computer composer interfaces. Interface designs can exploit the composers' already acquired auditory familiarity with musical instruments. For example, in the case of the *KEYed* user interface, by mapping piano notes to composition software functions, a unique timbre, along with the piano notes, can act as a mnemonic aid for the composer as to which note performs which functions, and also act as a feedback mechanism for reducing mode errors and reinforcing the system state. However, we also anticipate potential auditory interference problems which may arise as the earcons used in our design, which are basically sounds from the piano keyboard, might confuse a composer while he is composing on the same or different musical scales. The results of our studies when the earcons are used in the *KEYed* user interface prototypes are discussed in detail in Chapter 5. Finally, we believe that the careful and creative use of such auditory feedback can facilitate the control and use of the music composition workstations by composers with visual disabilities.

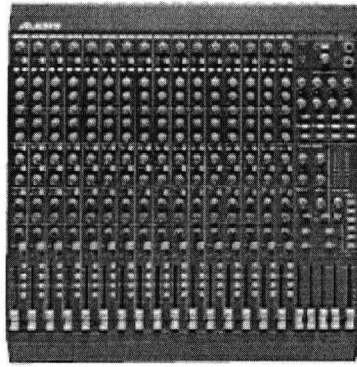
In the following sub-section we discuss a key design tradeoff which exists in the design of user interfaces in general. We begin by furnishing the two broad classifications of

user interfaces based on how the interfaces multiplex functions. Further, we discuss the importance of drawing a line between these two kinds of designs and its relevance in the design of composer computer interfaces.

2.2.5 Space - Time Multiplexing Tradeoffs in Input devices

Input devices can be broadly categorized as being space multiplexed or time multiplexed. With a space-multiplexed input, each function to be controlled has a dedicated transducer, each occupying its own space. For example, inside an automobile there is a brake, a clutch, a steering wheel, gearshift and so on, each used for different functions. However, time-multiplexing input uses one device to control different functions at different points in time. A typical example would be a computer mouse used for menu selection to pointing, to scrolling, and so on, at different times.

This is an important design decision to make in designing user interfaces as they both have their pros and cons, depending on what application the devices are used for. For example, a dedicated physical input device for every function can be costly and inefficient. Figure 2.3 in the following page shows a how a space multiplexed audio mixing console has hundreds of physical transducers to control individual functions, and a simple single transducer computer mouse which is time multiplexed.



(a) Audio mixer (space multiplexed)



(b) Computer mouse (time multiplexed)

Figure 2.3: Space Multiplexed Mixer (a), Time Multiplexed Mouse (b)

To relate this example to the composer computer interface design, let us consider this audio mixer and the mouse as the individual input devices being connected to music production software to manipulate the virtual software functions, such as faders and knobs. Though the audio mixer may look like a natural input device for these tasks, it can be costly and space consuming. At the same time, while the mouse is less expensive and less space consuming, it is not natural to turn knobs with a mouse considering the complex graphical user interfaces used in modern music production software.

In the study of graspable user interfaces, specialized physical form factors are used as input devices. Graspable user interface research [21] suggests that the ultimate benefits lie somewhere in between these two above extremes. Their experiments show that a space-multiplex input scheme with specialized devices can outperform a time-multiplex input design for certain situations. They found that the inter-device switching cost may not be as costly as they had anticipated, and that it may be faster to acquire an attached device that is out of hand than to attach to virtual controls with a device in hand. It is suggested that the

‘universal setup’, which is the keyboard and mouse, seems inefficient for users who work in a specific domain.

In the current music workstation setup, composers use a combination of the keyboard (mostly space-multiplexed) and the mouse (time-multiplexed) to control various software functions. The *KEYed* user interface incorporates a space multiplexed design facilitated by the piano keyboard. Each piano key facilitates the control of a specific virtual function of the software, which is further laid out in an octave structure where the different octaves on the piano are mapped to specific windows in the software. The effectiveness of such a design in the *KEYed* user interface is discussed in detail in Chapter 5.

In the following concluding sub-section, we begin by studying the human manual capabilities and capacities in tasks involving different roles assigned to the two hands. Further, we discuss about the importance of bandwidth [25] while assigning such tasks, the ways to predict and reduce movement time in manual activities, and how we apply these issues in the design of the *KEYed* user interface.

2.2.6 Human Manual Studies

Most human everyday manual activities fall into the following three classes:

1. Bimanual and Asymmetric activities (for example, playing a piano).
2. Unimanual activities (for example, throwing a ball) [22].
3. Bimanual and Symmetric activities (for example, weightlifting with a bar).

It is emphasized that in humans, the most skilled manual activities involve two hands playing different roles. Guiards’ Kinematic Chain model on human bimanual action strongly suggests that the two human hands work in a cooperative and asymmetric manner. Further, the dominant hand tends to act later, work in a smaller but finer scale, and operate

within the frame of reference provided by the non-dominant hand [22]. This is evident as shown in studies where two hands engaged in different motor control mechanisms (one in an isotonic position control and one in an isometric rate control) outperformed the one handed conditions for browsing tasks [27]. For more demanding tasks, such as a graphical mail sorting task in which the users need to drag a mail icon into a folder window, scroll the window while keeping the dragged objects, and then dropping it in an intended folder, they observed more advantages with the two handed system than the one handed condition. Other important studies in bimanual actions have shown that when a user can manipulate the entire object or function with two hands as an integrated chunk that is greater than the separate elements, as in an unimanual input, both manual (the elimination of repeated control points reacquisition) and cognitive (the reduction of mentally visualizing the control points) advantages are gained [23].

Composers, due to their familiarity with musical instruments, are very skillful in bimanual actions. For example, while playing a stringed instrument, musicians assign completely different roles to their hands. Beyond manual actions, they even assign tasks such as latching effects, processing with their feet while playing a guitar, or sustaining notes with a foot pedal while playing a piano. An extreme case would be a drummer who assigns different roles to both his hands and feet at the same time. In other words, composers are quite skillful and comfortable in performing different functions simultaneously by assigning the functions to different body channels.

Considering this, in our design of the *KEYed* user interface, it is natural for us to take advantage of these results from bimanual studies. An example of a bimanual interaction found in the *KEYed* user interface involves assigning a single-point-touchpad to

one hand for performing a full two degrees of freedom (DOF) task, such as drawing a modulation graph while holding a piano note with the other hand, as discussed in Chapter 1. The non-dominant hand set the reference for the dominant hand operations, and the sustained engagement of the reference reduced the micro mode errors discussed in the Sub-section 2.2.3. Hence, composers performed these bimanual tasks quite effectively as discussed in Chapter 5.

In such task assignments, it is also critical to understand the importance of the term 'bandwidth' [25].

Fitts [24] proposed a formal relationship to describe human unimanual performance in aimed movements by borrowing literature from information theory. The following equation shows one formulation of Fitts' law:

$$T = a + b ID; \text{ where } ID = \text{Index of difficulty} = \log_{\text{base } 2} (A/W + 1)$$

Fitts' law predicts that the time needed to point to a target of width W at a distance A is T seconds, where b is the slope, and a is the intercept. The capacity for executing a particular class of motor responses in bits per second, or the channel bandwidth, takes the following relationship:

$$IP = ID / T; \text{ where } IP = \text{Index of performance or channel bandwidth}$$

Intuitively, the higher the bandwidth, the higher the rate of human performance since more information is being articulated per unit time. One of the strengths in Fitts' law is that the measures for IP , or bandwidth, can motivate performance comparisons across factors such as device, limb, or task [25]. Therefore, operations using the KEYed user interface can be optimized by selecting and combining conditions yielding a high bandwidth. This also facilitates a direct comparison of the interface devices used for

composing tasks (example: touchpad versus mouse for a volume fading task), considering the bandwidth of the body channels involved (example: index finger on a touchpad versus sliding a knob with two fingers).

Focusing on the bandwidth of the body channels, neurophysiological studies have shown that various parts of the human body are anatomically reflected in their brain to their physical size and mass. Representations of the fingers and the hands are much richer than those of the wrists, elbows and shoulders, and studies since have shown significant performance enhancement if fine muscle groups (like fingers) are allowed to take part in handling an input device, due to their higher information processing bandwidth over other body parts [28] [29]. Information processing bandwidths for the fingers, wrist and arm are found to be 38 bits/sec, 23 bits/sec and 10 bits/sec, respectively, and are useful quantitative evidence for designing new interfaces, such as the *KEYed* user interface.

Another aspect of the manual activities while being engaged with the *KEYed* user interface is the movement time. The composer's arms, wrists and fingers busy themselves on the piano keyboard, computer keyboard and mouse. Different models of manual movements for a single device acquisition and inter device movements can be used as an integral part of the user interface design. Fitt's law is a powerful model for the prediction of movement time in human computer interaction [25]. Useful movement time predictions are done on various input devices to perform tasks of varying *A* (Amplitude) and *W* (Width) conditions. A more appropriate model for the *KEYed* user interface is the Key Stroke Level Model. The Keystroke Level Model (KLM) [39] proposed by Card, Moran and Newell (1983), predicts the task execution time from a specified design and specific task scenario. The KLM is a simplified version of GOMS [38] in that it focuses on very low level tasks.

The actions are termed keystroke level if they are at the level of actions similar to pressing keys, moving the mouse, pressing buttons, and so forth.

The physical operations that are performed by users are the following:

K - keystroking, actually hitting keys on the keyboard

B - pressing a mouse button

P - pointing, moving the mouse or other device to a target

H - homing, switching the hand from the mouse to the keyboard or vice versa

M - mentally preparing for a physical action

R - system response, this may be ignored if the user does not have to wait for it.

Thus, each task is broken into the sequence of the above operators. The **M** operator is not meant to represent cognitive thinking, but merely the recall that humans do when preparing to do expert tasks. The **P** operation is based on Fitt's law which states that the time to point to a certain item depends on the distance and size of the item. The individual times for each of these tasks could be dependent on the users, the hardware, and the application. Thus, the averages would have to be determined by experimentation. The estimated times for these operators during a typical computing task [38, p.264] is given in the following page:

Operation Name	Operation Symbol	Time (sec)	Notes
Keystroke	K	0.20	Average skilled typist (55 wpm)
Pressing Mouse Button	B	0.10	
Pointing	P	1.1	This is an average recommended estimate [41]. To use Fitt's model for actual time.
Homing	H	0.40	
Mental Act of Perception	M	1.2	This is an average recommended estimate [41].

Table 2.1: Estimated Times for the K, B, P, H and M Operators

Analyzing such results from performing the KLM activity could suggest ways to improve the user interface by showing what tasks or operators are taking the most time. By reducing the number of operators, or the time for a particular operator the performance of the *KEYed* user interface could be improved. Note that for the current music workstation setup, the estimated time for the **H** operator is more than the ones estimated in Table 2.1, due to the keyboard / mouse positioning as shown in Figure 4.3. For predicting movement times in music workstations, the **H** operator should also be refined and further classified as follows:

H1: Homing between the piano keyboard and the computer keyboard or vice versa

H2: Homing between the computer keyboard and mouse or vice versa

H3: Homing between the piano keyboard and mouse or vice versa

In the interaction with the *KEYed* user interface, the homing times denoted by the operator **H1**, **H2** and **H3** are zero, due to its design as discussed in Chapter 3. However, there is homing involved between the piano keyboard and the touchpad in the prototype II as we will discuss in Chapters 3, 4 and 5. The predicted KLM task completion times for the various tasks used in our studies are further discussed in the Chapter 4. Comparing such predicted times to the measured task completion time could suggest ways to improve the interface designs discussed in the Section 2.1 by giving insights into the time required for the various operators involved when using such interfaces.

2.3 Summary

In summary, we began this chapter by discussing some alternative composer computer interface designs, recently introduced by music system manufacturers and researchers. We see how significant user interface research exists in the field of music synthesis and performance, where the interface is used on-line [38], as opposed to music composition workstations which researchers consider as more of an off-line editing tool. Considering these possibilities, we discuss in detail, how the principles and findings in the field of Human Computer Interaction (HCI) can be applied to the design of composer computer interfaces. At various design stages of the *KEYed* user interface we gave careful attention to these findings. The following chapters furnish the design and evaluation of the different *KEYed* user interface prototypes in this design space.

Chapter 3

KEYed User Interface

We begin this chapter by introducing the two prototypes of the *KEYed* user interface we built and further, in Section 3.3 we discuss the system design and implementation in detail.

3.1 KEYed User Interface Prototype I

The *KEYed* User Interface Prototype I was built for the purpose of studying both the initial reaction of composers and the effectiveness of the new interface for digital music workstations. The results of this study are detailed in Chapter 4.

The music production software used in this prototype is the Cubase® 5.0 VST® 32 by Steinberg® for Microsoft® Windows® 98 systems, which is a popular setup among composers. The software for this prototype is written to comply with the Windows® 98 standards. Macros from the different Cubase® VST® windows are mapped to specific octaves on the piano keyboard for separation. The piano keyboard mappings-to-functions used in this prototype are common to both prototypes and are discussed in detail in Section 3.3. The single-point-touchpad discussed earlier is not integrated into this prototype. The piano keyboard used in this prototype, which is shown in Figure 3.1, does not have multiple foot switch ports. Typically, composers use a ‘sustain pedal,’ which is a momentary foot switch as well, to sustain musical notes. Hence, studies with this system do not deal with potential mode errors that occur due to switching between the *KEYed* foot switch and the sustain pedal while composing and editing.



Figure 3.1: KEYed User Interface Prototype I

3.2 KEYed User Interface Prototype II

A more advanced piano keyboard, like the ones typically used by composers, is utilized in the KEYed User Interface Prototype II. Such piano keyboards have multiple foot switch ports and more piano keys, as shown in Figure 3.2, hence facilitating more keys to function mappings and the use of multiple foot pedals.

The music production software used in this prototype is the Nuendo® 1.5 for Microsoft® Windows® 2000 systems, which is a recent music production software package released by Steinberg®. The software for this prototype is written to comply with Windows® 2000 standards; the design is discussed in Section 3.3.



Figure 3.2: KEYed User Interface Prototype II

The single-point-touchpad used in this prototype is the Cirque® glide point pad (Width: 65mm, Height: 49mm). A plexi glass framework is built around the touchpad for protection, and its sides are filed to round out sharp edges, taking into account tasks which require scrolling at the sides, as shown in Figure 3.3.

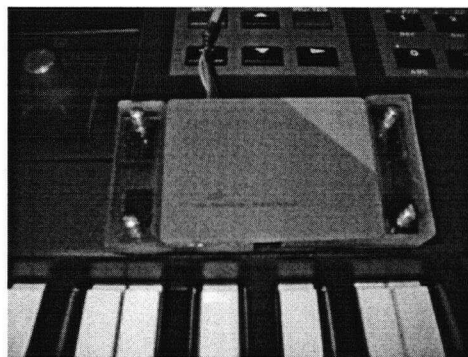


Figure 3.3: Single-Point Touchpad

In the following section we discuss the *KEYed* user interface system design in detail.

3.3 KEYed User Interface System

The *KEYed* user interface system design and implementation section is subcategorized into the following:

- 3.3.1 Hardware Layout
- 3.3.2 Piano Keyboard Mapping Design
- 3.3.3 Software Design

3.3.1 Hardware Layout

Figure 3.4, shown below, is the layout of the *KEYed* User Interface prototype I hardware. The piano keyboard is connected to the sound cards thru the Music Instrument Digital Interface (MIDI) ports for transferring performance data to the computer. The MIDI THRU port on the piano keyboard is basically an 'echo' of the MIDI IN port.

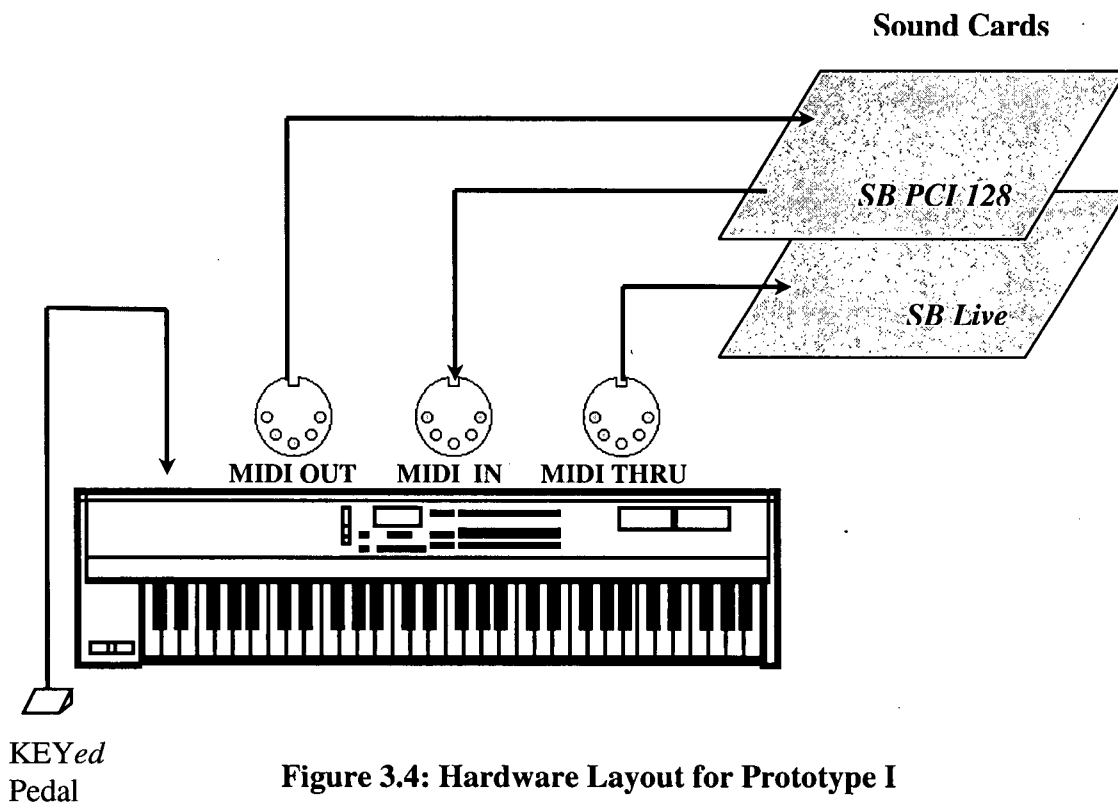
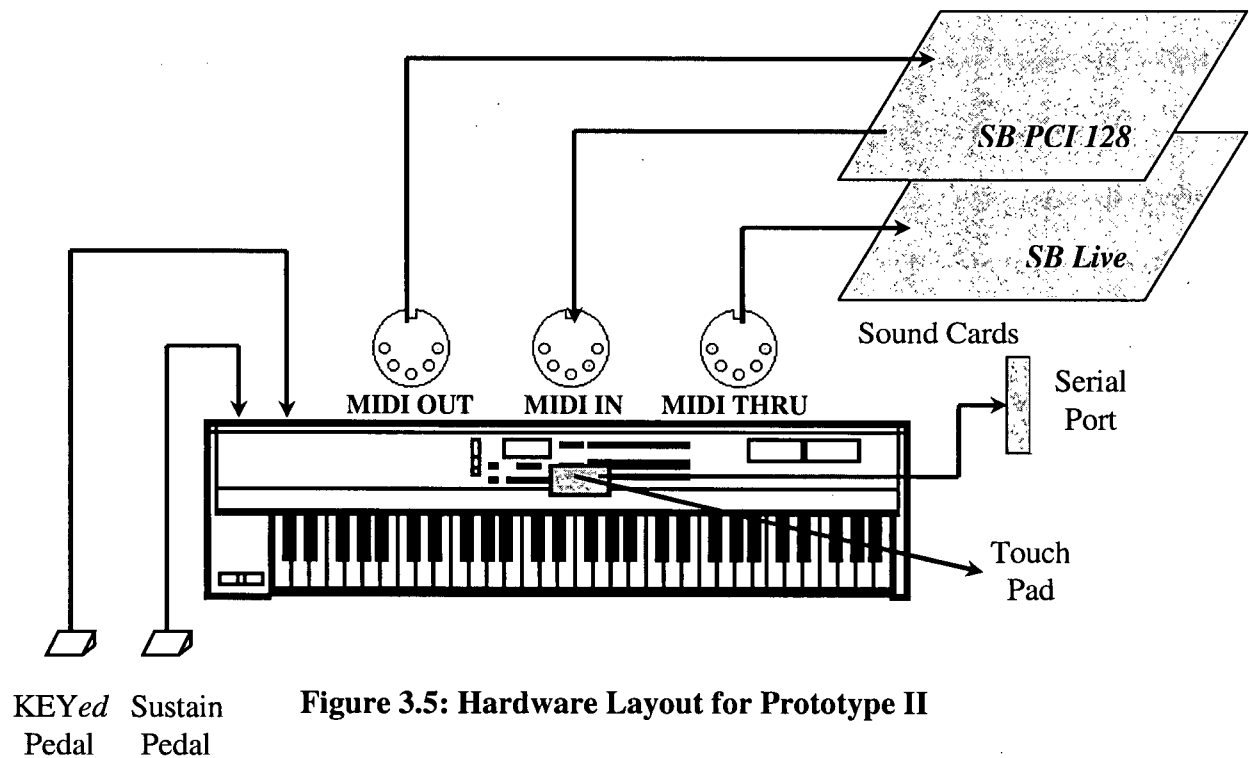


Figure 3.4: Hardware Layout for Prototype I

An introduction to the MIDI protocol is discussed in Section 3.3 on software design. The MIDI data from the piano keyboard is sent to the sounds cards, Sound Blaster® Live® and Sound Blaster® PCI 128, as shown in Figure 3.4. A momentary foot switch, labeled *KEYed Pedal*, is used in this system as a mode switch because it is a familiar device in this domain, hands free, and provides appropriate feedback. The piano keyboard used for the prototype II has multiple foot switch ports, thereby facilitating the use of the sustain pedal along with the *KEYed* pedal. Further, the single-point touchpad used in prototype II interfaces with the serial port, as shown in Figure 3.5.



3.3.2 Piano Keyboard Mapping Design

As discussed earlier, we use Cubase® 5.0 VST® 32 and Nuendo® 1.5 music production software by Steinberg® on Microsoft® Windows® for our prototypes. Macros from the different windows from the software are mapped to specific octaves on the piano keyboard for separation. The windows chosen for this mapping are common to the set of software used for our prototypes, as well as to other popular music production software, including Logic Audio® and Digital Performer®. Hence, the following discussion on piano keyboard mappings applies to most music production software.

The design of the layout incorporates the windows in the production software most commonly used by the composer. This is identified in our requirements analysis to reveal the commonly used and useful music workstation functions, while composing with the Cubase® and Nuendo® software. The functions associated with each window map to a

specific octave on the piano keyboard. Further, the octave C6 to B6 of the piano keyboard is used for general editing functions, such as cut, copy and paste of selected parts.

The design of the piano keyboard mappings used for our prototypes has been laid out as follows, although they can be reconfigured or transposed. The octaves used are the central ones found on every standard piano keyboard. The mappings of the windows to the octaves are shown in Table 3.1.

WINDOW	OCTAVE USED
Mixer / EQ Windows	C2 to B2
Transport Window	C3 to B3
Arrange Window	C4 to B4
Key Edit Window	C5 to B5
Edit / File	C6 to B6

Table 3.1: Window - Octave Layout

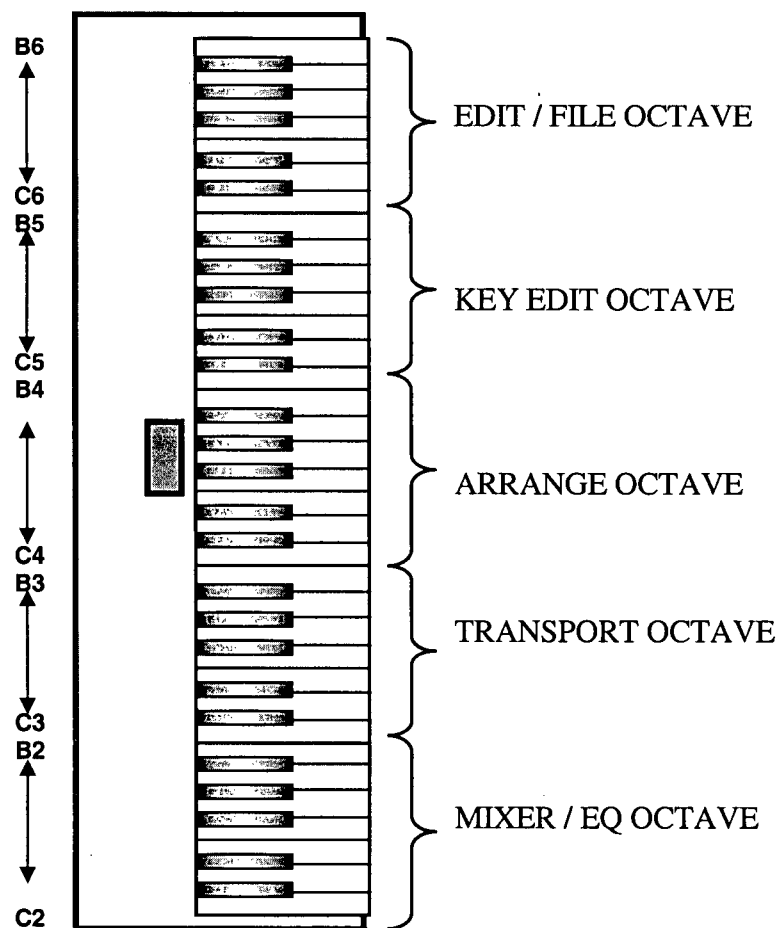


Figure 3.6: KEYed User Interface Piano Mappings

The names used to identify the piano keys, such as 'B4' for example, are standard, set by the MIDI Manufacturers Association. Figure 3.6 below shows the octave layout. Note, we have named the octaves based on the windows they are mapped to. The detailed mappings of the different octaves are as follows:

a) Mixer / EQ Octave (C2 to B2)

The Mixer / EQ octave is used for windows which allow for tasks such as volume mixing, effects and dynamic processing. Common layouts of these windows consist mostly of knobs and faders for controlling continuous parameters.

To perform such tasks we use a single-point touchpad, as shown in Figure 4.3. While the touchpad is a new device to the composer, we placed it carefully to minimize the device acquisition time. The touchpad is used for constrained vertical or horizontal actions, for a single degree of freedom (DOF) task, such as sliding a fader or turning a knob, and to perform a full 2DOF task, such as drawing a modulation graph.

Different notes-to-function mappings used in the *KEYed* user interfaces' mixer / EQ octave are shown in Table 3.2, along with the current method of working with these functions, on the following page.

With KEYed User Interface		Without KEYed User Interface
Note	Mapped Function	
C2	Mixer Open	F3
C#2	Select Master Fader	Mouse
D2	Select EQ	Mouse Click
D#2	Select Frequency 1	Mouse Click
E2	Select Gain 1	Mouse Click
F2	Select Gain 2	Mouse Click
F#2	Select Frequency 2	Mouse Click
G2	Select Gain 3	Mouse Click
G#2	Select Frequency 3	Mouse Click
A2	Select Gain 4	Mouse Click
A#2	Select Frequency 4	Mouse Click

Table 3.2: Mixer / EQ Octave Layout

These mappings are laid out in Figure 3.7.

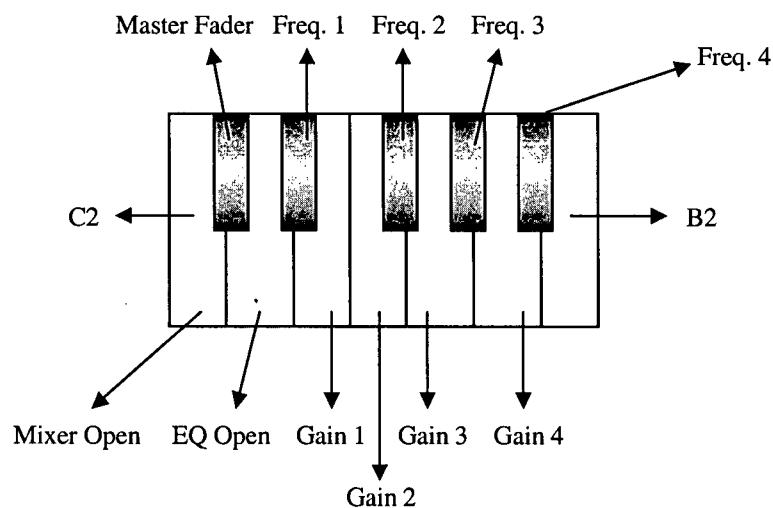


Figure 3.7: Mixer / EQ Octave Layout

Here is an example of how the piano keys and the touchpad work in conjunction in this octave:

1. While the KEYed foot pedal is depressed, the composer can open the mixer window by holding the note 'C2' on the piano keyboard.

2. By selecting the note 'C#2' while holding down the first note 'C2', the composer can then select the mixing fader in the mixer window.
3. While holding the notes 'C2' and 'C#2', a single DOF fading task can be performed using the single point touchpad.

The touchpad facilitates performing continuous parameter control with the selected knobs and faders. Very fine controls of knobs and faders are possible with the single-point touchpad used in our design.

Notice, with such a design, a double note combination, or playing a chord selects the specific mixing fader of interest directly, while the KEYed pedal is depressed. Figure 3.8 shows a fading task being performed with the touch pad.

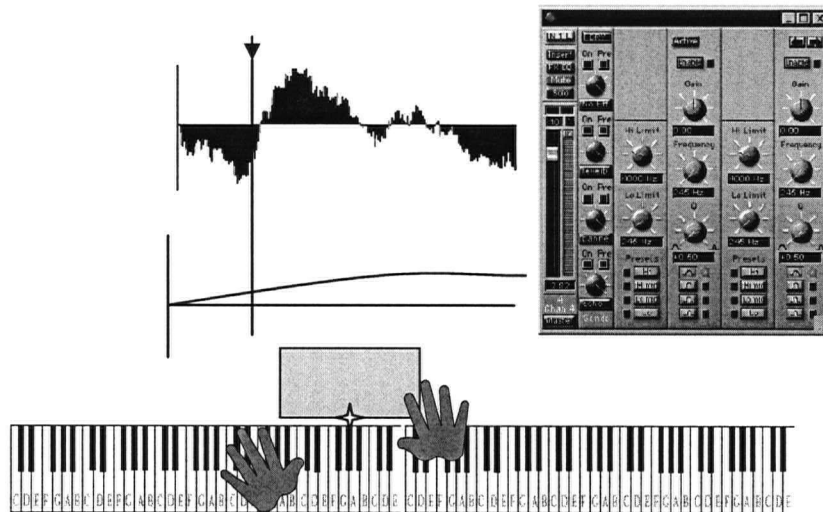


Figure 3.8: Editing with Mixer / EQ Octave

b) Transport Octave (C3 to B3)

The transport octave is used to play, stop, record, mute, and solo a sequence or a specific channel. It is the window most frequently used by the composer. Some of the most commonly used transport window functions by the composer are mapped to the piano

keyboard. Different notes to function mappings used in the *KEYed* user interface transport octave are shown in Table 3.3, along with the current method of working with these functions.

With <i>KEYed</i> User Interface		Without <i>KEYed</i> User Interface
Note	Mapped Function	
C3	Open Transport	F12
C#3	Stop	0 (Numpad)
D3	Play	P
D#3	Record	R
E3	Rewind	4 (Numpad)
F3	Punch Out	O
F#3	Punch In	I
G3	Forward	6 (Numpad)
A3	Return to 0	1 (Numpad)
B3	Click	M

Table 3.3: Transport Octave Layout

These mappings are laid out in Figure 3.9.

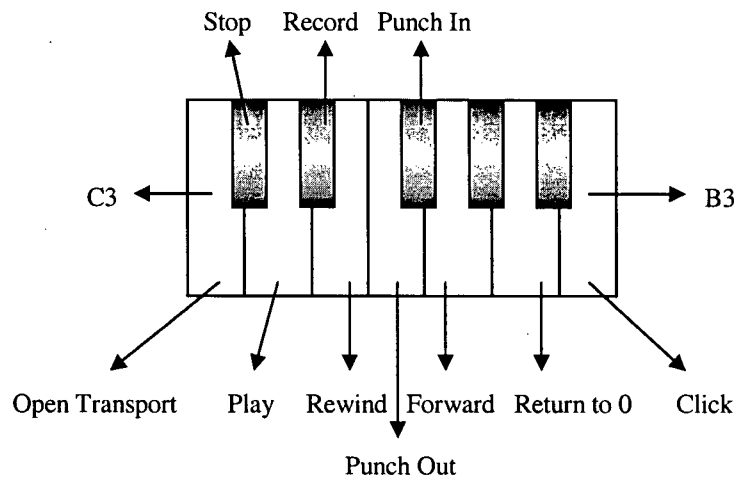


Figure 3.9: Transport Octave Layout

Here is an example of how the piano keys can be used in this octave:

1. While the *KEYed* foot pedal is depressed, the composer can open the transport window by holding the note 'C3' on the piano keyboard.

2. By selecting the note 'D3' while holding down the first note 'C3', the composer can play the selected track.

The transport window used in the Cubase 5.0 VST program is shown below in Figure 3.10, along with examples of the mappings.

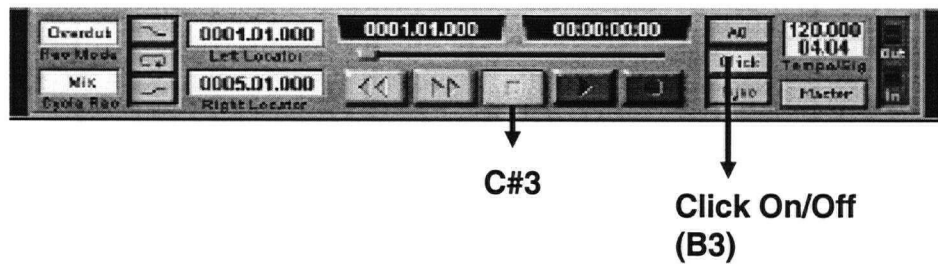


Figure 3.10: Transport Window

c) Arrange Octave (C4 to B4)

The arrange window is used by the composer to create and arrange tracks during composition. Frequent operations include toggling between tracks and sections within the tracks, selecting and moving these sections to desired positions, and so on. Examples of the functions and their octave mappings used in the *KEYed* user interfaces' arrange octave are shown in Table 3.4, along with the current method of working with these functions.

With <i>KEYed</i> User Interface		Without <i>KEYed</i> User Interface
Note	Mapped Function	
E4	Select previous part	Left arrow
F4	Select next part	Down arrow
F#4	Select upper part	Up arrow
G4	Select lower part	Right arrow

Table 3.4: Arrange Octave Layout

Notice that the middle four keys (E, F, F# and G) of the Arrange Octave are used for toggling between the tracks or sections. They correspond to the computer commands '←', '↓', '↑', '→' and are visually similar in their arrangement. These mappings are laid out in Figure 3.11.

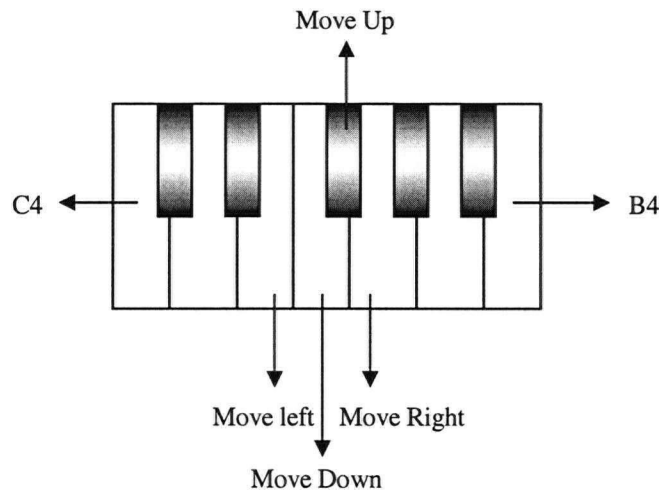


Figure 3.11: Arrange Octave Layout

Here is an example of how the piano keys can be used in this octave:

1. While the *KEYed* foot pedal is depressed, the composer can toggle up and down the track list by selecting the notes 'F#4' and 'F4' on the piano keyboard.

The arrange window used in the Cubase 5.0 VST program is shown below in Figure 3.12.

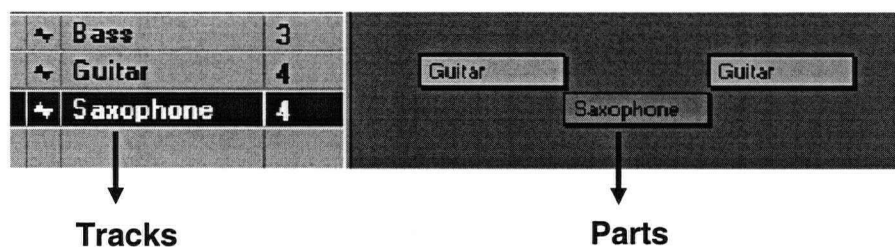


Figure 3.12: Arrange Window

d) Key Edit Octave (C5 to B5)

The Key Edit window is used by the composer for editing notes. Typically, this window is opened by double clicking on the section of the track to be edited, or by using a computer keyboard command after selecting the section. Various parameters of the individual notes in that section, including the length of notes, velocity, and pan, can then be edited. Examples of the functions and their octave mappings used in the *KEYed* user interfaces'

key edit octave are shown in Table 3.5, along with the current method of working with these functions.

With KEYed User Interface		Without KEYed User Interface
Note	Mapped Function	
C5	Open Key Edit	E
C#5	Draw notes or graph	Mouse Click
E5	Previous note	Left arrow
G5	Next note	Right arrow

Table 3.5: Key Edit Octave Layout

These mappings are laid out in Figure 3.13.

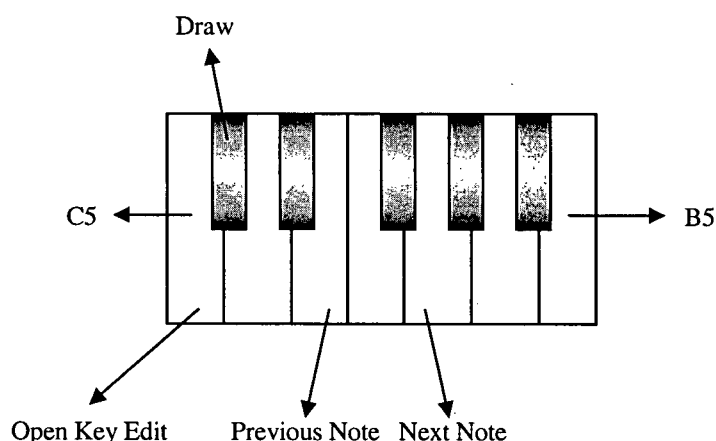


Figure 3.13: Key Edit Octave Layout

Here is an example of how the piano keys can be used in this window:

1. While the KEYed foot pedal is depressed, the composer can open the key edit window by holding the note 'C5' on the piano keyboard.
2. By selecting notes 'E5' and 'G5' while holding down the first note 'C5', the composer can toggle back and forth between the individual notes of interest within the key edit window. The arrange window used in the Cubase 5.0 VST program is shown below in Figure 3.14.

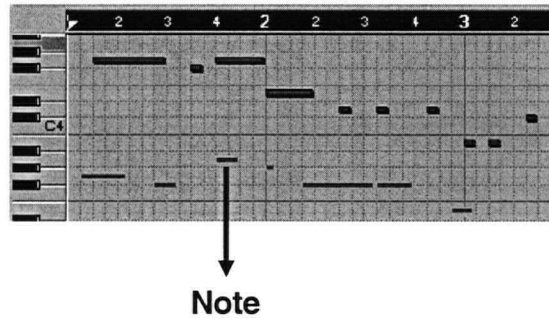


Figure 3.14: Key Edit Window

3. Further, holding the notes 'C5' and 'C#5' simultaneously results in the selection of the drawing tool. A volume graph can then be drawn using the touchpad, or a note can be drawn on a specific section of the score as shown in Figure 3.15.

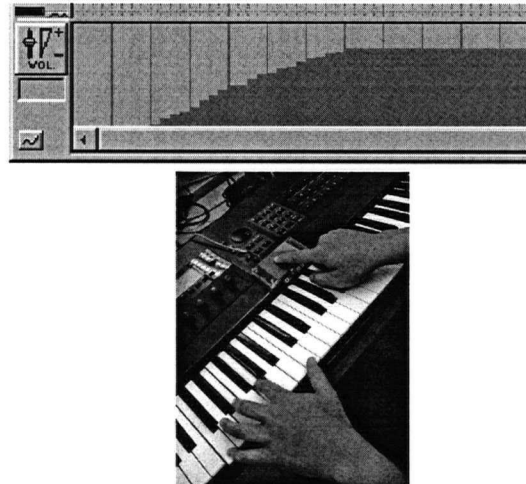


Figure 3.15: Drawing Volume Graphs

e) Edit / File Octave (C6 to B6)

This octave allows the musician to do basic editing and file handling functions, such as copy, paste, open file window and so forth from the piano keyboard by pressing the *KEYed* foot pedal. This octave is not mapped to any specific window on the software program. Examples of the functions and their octave mappings used in the *KEYed* user interfaces'

edit / file octave are shown in Table 3.6, along with the current method of working with these functions.

With KEYed User Interface		Without KEYed User Interface
Note	Mapped Function	
C6	Open file	F
D6	Close file window	Esc
E6	Cut	X
F5	Copy	C
F#6	Undo	Z
G6	Paste	V

Table 3.6: Edit / File Octave Layout

These mappings are laid out in Figure 3.16.

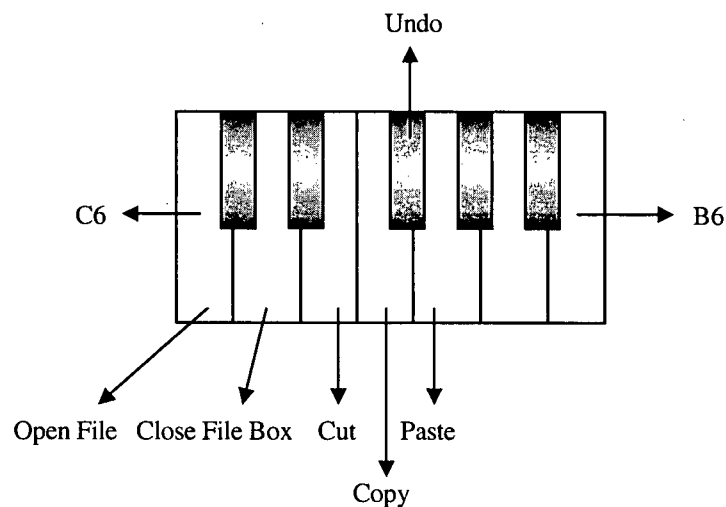


Figure 3.16: Edit / File Octave Layout

Further, most mappings are also designed to take into consideration the timbre associated with the notes and their combinations. Although there is no formal model used for the timbre mapping, interesting mappings are achieved. For example, to 'Forward' and "Rewind" the arrange window time line to a specific spot on the window, the composer

'Opens' the transport window ('C4'), 'Forwards' the time line ('G4') and 'Rewinds' the time line ('E4') to the desired spot. This resembles a C Major musical chord structure.

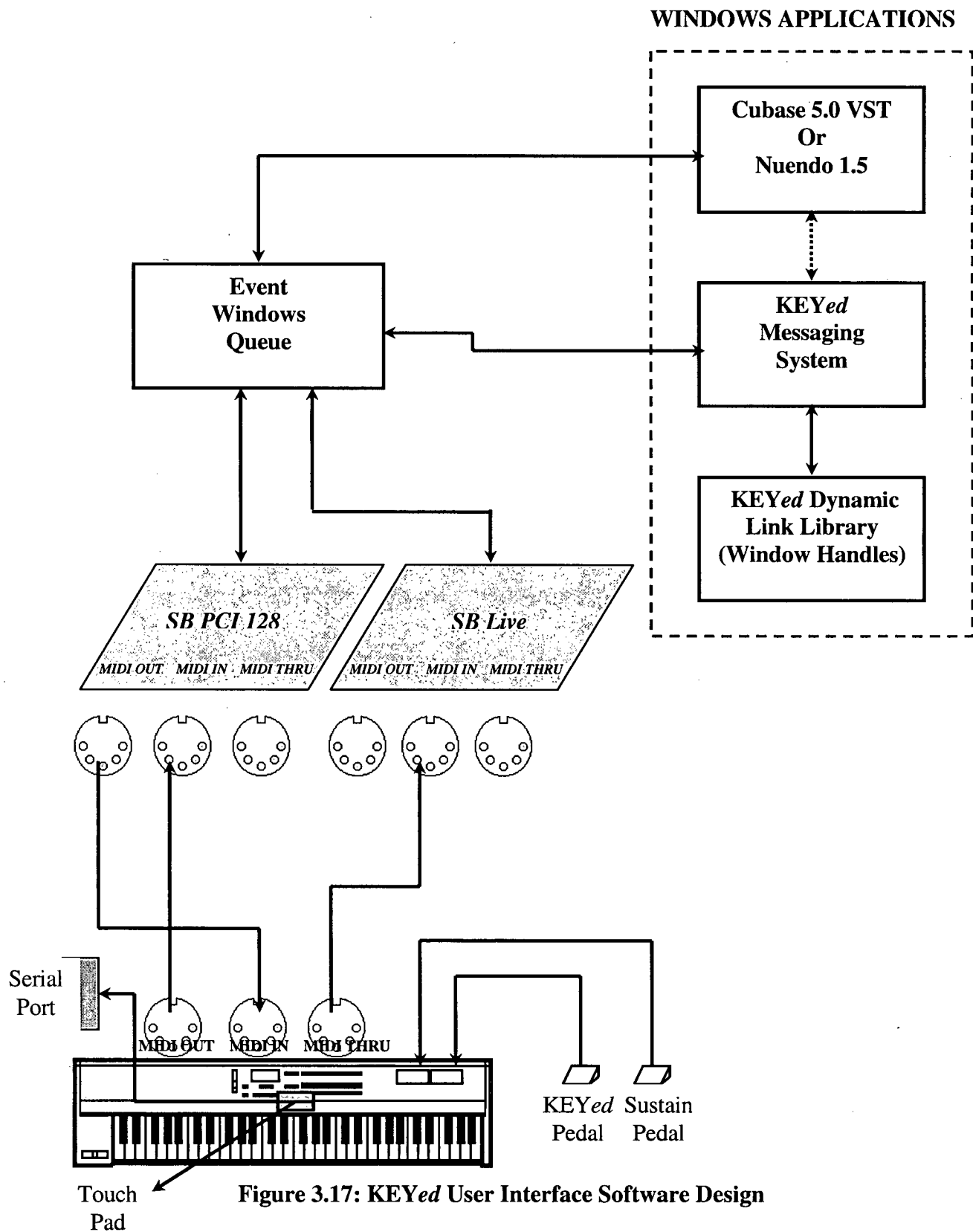
The key combinations are also designed to retain the composers' typical fingering patterns on a piano keyboard. The highest note, 'G7', of the piano keyboard used in prototype II, is used as an 'Enter' key for all confirmation functions. We anticipate that composers can conveniently hit this key, as it is at the extreme end, and hence, doesn't need to be looked for. The note 'F#7' is mapped to the Steinberg's help page for composers.

3.3.3 Software Design

The basic requirement for the software is as follows:

Whenever the *KEYed* pedal gets activated, the music production application responds to the MIDI input from the piano keyboard, as though it were computer keyboard and mouse messages.

To achieve this, the *KEYed* software intercepts all the MIDI data entering the MIDI device ID's of the sound cards and sends the appropriate windows messages to the target application, whenever the *KEYed* pedal gets activated. The intercepted MIDI messages are serial. The bits are generated at a rate of 31,250 per second, but it takes ten bits to make a character and up to three characters to make a message, so it takes most of a millisecond to send a message. Each action taken on the piano keyboard (such as releasing a key) generates a message. The typical message contains a channel number, a code for the key or other control affected, and descriptive data, such as key velocity. The software layout of the *KEYed* user interface system is shown in below in Figure 3.17.



The core of the software consists of the *KEYed* Messaging System, written in Visual Basic, and the *KEYed* Dynamic Link Libraries written in C. The *KEYed* pedal, which is plugged into the foot switch port of the piano keyboard, takes the value MIDI controller #4, and the sustain pedal takes the value MIDI controller #64. When an activation of the controller #4 message is read by the *KEYed* Messaging System from the Event Windows Queue, it posts messages to specific windows within the music production software (Ex. Nuendo), based on the MIDI note messages received after that point, and their mappings as discussed in the previous Sub-section 3.3.2.

The *KEYed* Dynamic Link Library program which gets linked at run-time with the *KEYed* Messaging System, updates the handles to the different windows in the application as the composer works, and further posts certain mouse cursor messages.

The touch pad used in our design is a standalone unit built by Cirque Inc., which connects through a serial port. By disabling clicking functionalities from the touchpad and by simulating mouse button holds and clicks on the piano keyboard, the *KEYed* user interface system facilitates the use of piano keys for the selection of drawing tools and performing mouse button functions.

KEYed User Interface Experiments

Applying an iterative design approach in the development of the *KEYed* user interface, we conduct studies at two different stages of development, to test the effectiveness of the interface during digital music composition. A preliminary study with the *KEYed* user interface prototype I, and further, a more detailed experiment including a case study with our prototype II is done with composers. This chapter discusses the experimental designs used in these studies and the analysis of the derived results. A comparison of the predicted Keystroke Level Model (KLM) task completion times for the various tasks and the measures task completion means is furnished at the end of this chapter in Section 4.3. Discussions based on these results are given in Chapter 5.

4.1 Preliminary Studies with Prototype I

As mentioned in the Chapter 3, prototype I was built for the purpose of studying the initial reaction of the composers towards the *KEYed* user interface and its effectiveness in performing simple editing tasks during composition.

4.1.1 Goals and Hypothesis of the Study

In our experiment, two interaction methods are tested for their performance during two music editing tasks. A within-subjects design is chosen for this experiment to reduce subject variability. The interaction methods (levels) tested independently is as follows:

- 1) A Computer Keyboard
- 2) *KEYed* User Interface Prototype I with the sound from the piano turned on while editing.

As the *KEYed* system is a new interface in digital composition workstations, it is critical to study the initial reaction of the composers. Further, because the system design incorporates interaction styles already familiar to end users (composers), it is interesting to see whether the interface improved their practice with its current design.

In our experimental hypothesis, first, we need to demonstrate that *composers are able to perform simple, repetitive music composing and editing tasks with the KEYed user interface prototype I, faster than the computer keyboard used for the same set of tasks*. To achieve this, we optimize the positioning of the computer keyboard, and the macros used for the tasks, while using a computer keyboard, based on current practices. Second, we need to demonstrate that the *composers find the new interface easy and comfortable to use*. A questionnaire (Appendix A.1) is used to measure the composers' reactions.

4.1.2 Participants

A total of six composers with moderate to expert digital music workstation experience perform the tests with the two interaction methods (Computer keyboard and the *KEYed* user interface prototype I) in a pre-assigned order. A basic questionnaire, as shown in Appendix A.1, is used to study the subjects' background and their reaction to the initial system layout. The composers are not paid for their time, and recruiting is accomplished by word of mouth.

4.1.3 Controlled Experiment Design

First, as mentioned earlier, a within-subjects design is chosen for this experiment to reduce subject variability. The computer keyboard and the *KEYed* user interface prototype I (levels) are used by composers to perform two sets of tasks, independently. The composers perform the tasks and are tested for the task completion time (dependent variable). These

tasks labeled as Task A (Appendix A.2) and Task B (Appendix A.3), are explicit recipes for performing the different parts of the tasks in point form, provided to the composers for testing. As the touchpad is not integrated into this prototype, the task design takes into consideration only those functions that are commonly performed by composers during editing using the computer keyboard, and not the mouse. Each set of tasks involves repetitive composing and editing, thereby requiring the composers to switch back and forth between modes on their own. The individual tasks are repeated once to test for improvements in performance time. Some of the components of the tasks used in our experiments include opening windows, recording a section of track, soloing a specific track, creating new tracks, performing cuts and pastes of the recorded track, and so on. The order of the tasks and the independent variables used to perform the first set of tasks (Test1) and its repetition (Test 2) are randomized (counterbalanced) to avoid any learning confounds.

Subjects are given a 'practice run' to explore different modes and strategies. The learning time given to subjects for practicing with the *KEYed* user interface is measured. The piano keys and the octaves are explicitly labeled, as shown in Figure 3.4, to reduce learning time.

We choose to turn the sound 'on' while the piano keyboard is in the edit mode, in order to observe the effectiveness of the associated earcons [6], specifically to observe any improvements in performance time, while repeating tasks. After the tests, the subjects are asked to rate the input devices on a '0 (terrible) to 6 (great)' scale, based on their experiences. During testing, we explicitly check for any potential mode errors from the

KEYed foot pedal, while switching back and forth between the playing and edit modes, while using the KEYed user interface.

4.1.4 Analysis of the Results

The average learning time among the six composers with the KEYed user interface prototype I are measured to be approximately 5 minutes. A paired t-test analysis is performed on the mean completion times for Task A (Appendix A.2) and Task B (Appendix A.3) for the computer keyboard and KEYed user interface based interactions. The first set of tasks is labeled as Test 1 and the repetition of the tasks is labeled Test 2 in our analysis. Subjects perform both the tasks significantly better with the KEYed User Interface interaction in Test 1 ($p < 0.05$, t-Test), and not significantly, although faster, in Test 2 ($p < 0.06$, t-Test), when compared with the computer keyboard interaction. Test 2 is performed faster than Test 1, as shown in Figure 4.1. Subjects gave the computer keyboard interaction a significantly lower rating than the KEYed User Interface interaction ($p < 0.02$, t-Test) as shown in Figure 4.2.

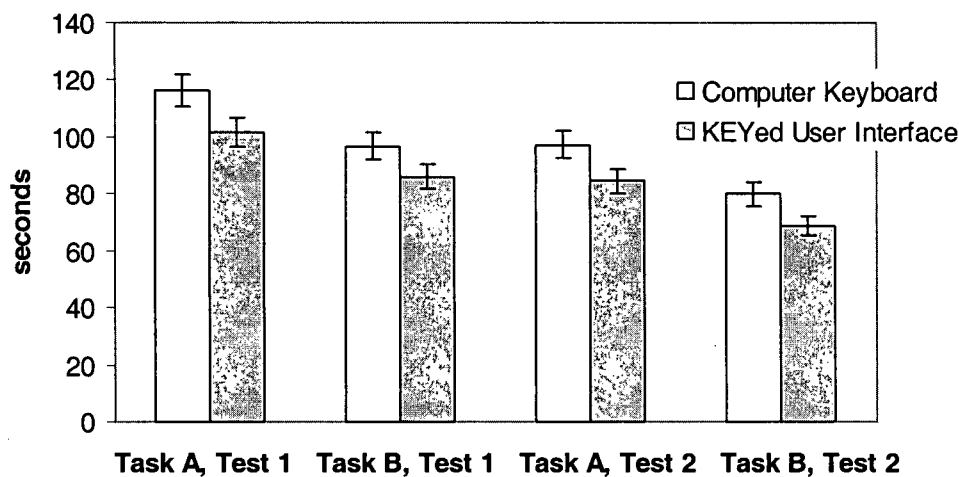


Figure 4.1: Mean Completion Times with 95% Error Bars

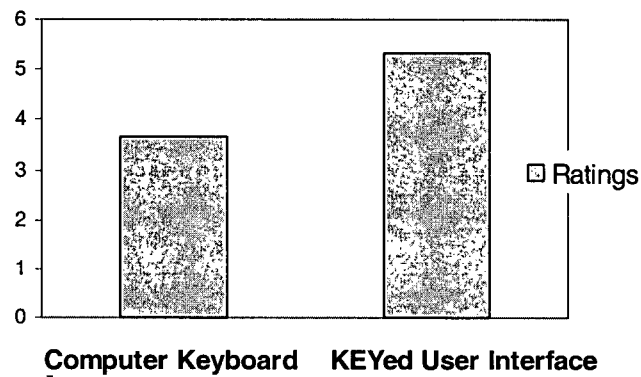


Figure 4.2: Mean Subjective Rating

Discussions based on these results and the findings from the provided questionnaire are furnished in Chapter 5, Section 5.1.

4.2 Final Studies with Prototype II

Based on these results from our Prototype I, we conducted studies on the effectiveness of the *KEYed* user interface prototype II. The studies consists of the following:

- a) A controlled experiment like the one used for prototype I, to study the effectiveness of prototype II for performing simple, moderate, and complex editing tasks during composition.
- b) A case study conducted in a home project studio over a one week period.

As explained in Chapter 3, prototype II includes a single-point touchpad for performing various tasks, such as drawing graphs and moving faders. All the keyboard mappings previously discussed in the Sub-section 3.3.2, are used in this prototype.

4.2.1 Goals and Hypothesis of the Studies

In our controlled experiment, four interaction methods (independent variable 1) are tested for their performance during three music editing tasks during composition. A within-subjects design is chosen for this experiment to reduce subject variability. The interaction methods (levels) tested independently is as follows:

- 1) A Computer Keyboard and Mouse combination
- 2) KEYed User Interface Prototype II with the sound from the piano turned off while editing.
- 3) KEYed User Interface Prototype II with the sound from the piano turned on while editing.
- 4) The single-point touchpad is tested exclusively for the tasks while being attached to the piano keyboard as shown in Figure 4.4.

Further, we perform a case study with the KEYed user interface prototype II to find any effects when the interface is used for longer periods of time.

In our experimental hypothesis with controlled experiments, first, we need to demonstrate that *composers are able to perform simple, moderate and complex repetitive music composing and editing tasks with the KEYed user interface prototype II, with the sound on the piano turned on or off, faster than the computer keyboard and mouse combination used for the same set of tasks*. To achieve this, we optimize the positioning of the computer keyboard and the mouse, and the macros used for the tasks while using a computer keyboard, based on current practices. The touchpad is carefully placed in the middle of the piano above the keys, shown in Figure 3.15. Our final experimental hypothesis demonstrates that *composers find the new interface easy and comfortable to use*.

Questionnaires A (Appendix A.4) and B (Appendix A.5) measure the composers' demographics, and their impressions before, during, and after the experiments.

4.2.2 Participants

The controlled experiments involve a total of ten composers with moderate to expert digital music workstation experience, performing three different tasks independently, with the four interaction methods in a pre-assigned order. The composers are not paid for their time. The case study is conducted with an expert composer using the *KEYed* system as part of his home project studio setup. This composer is not paid for his time either. All the recruiting was done by word of mouth.

4.2.3 Controlled Experiment Design

In this study, the composers are handed the consent form A (Appendix A.9) and detailed questionnaires before (Questionnaire A), during and after (Questionnaire B) the experiments. The purpose of questionnaire A, shown in Appendix A.4 is to collect demographic information to help illuminate the relationship between the subjects and the responses they make during the study. The purpose of questionnaire B shown in Appendix A.5 is to collect the composers' impressions of the experimental tasks, and of their performances.

The composers perform three different tasks (independent variable 2) and are tested for the task completion / reaction time (dependent variable). These tasks, handed to the composers and labeled Task A (Appendix A.6), Task B (Appendix A.7), and Task C (Appendix A.8), are simple, moderate and complex tasks, respectively. These tasks are not explicit recipes, like the ones used in the experiments with prototype I, but are stated in more general form. Here's an example: *On the current MIDI track, perform a recording by*

playing the piano for the pre-set number of bars with the sustain pedal pressed continuously during the recording.

The task design takes into consideration those functions commonly performed by composers during editing using the computer keyboard and the mouse, as the touchpad is integrated into this prototype. Each set of tasks involved repetitive composing and editing, thereby requiring the composers to switch back and forth between the modes on their own. Some of the components of the tasks used in our experiments include opening windows, recording a section of track, moving a fader to a specific amplitude, soloing a specific track, creating new tracks, turning a specific EQ knob to a particular frequency, drawing volume graphs, cutting and pasting a section of the recorded track, and so on. The order of the three different tasks and the four different independent variables used to perform the tasks are randomized (counterbalanced) to avoid any learning confounds.

Subjects are given a practice run to explore different modes and strategies. The learning time given to the subjects for practicing with the *KEYed* user interface is measured. The piano octaves are explicitly labeled as shown in Figure 4.4, to reduce learning time. The keyboard and mouse positioning are left to the discretion of the subjects.

During the experiments, the software checks for any potential mode errors from the *KEYed* foot pedal, while switching back and forth between the playing and editing modes, and while using the *KEYed* user interface. Since the sustain pedal is also used during tasks, mode error checking is also performed while the composer switches between the sustain and *KEYed* foot pedals. The different pedals are labeled as shown in Figure 4.4. Figure 4.3 shows the 'user studies booth' setup including one choice of keyboard and mouse positioning.



Figure 4.3: Prototype II Experiment Setup (Note: The user was free to reposition the mouse and the computer keyboard)

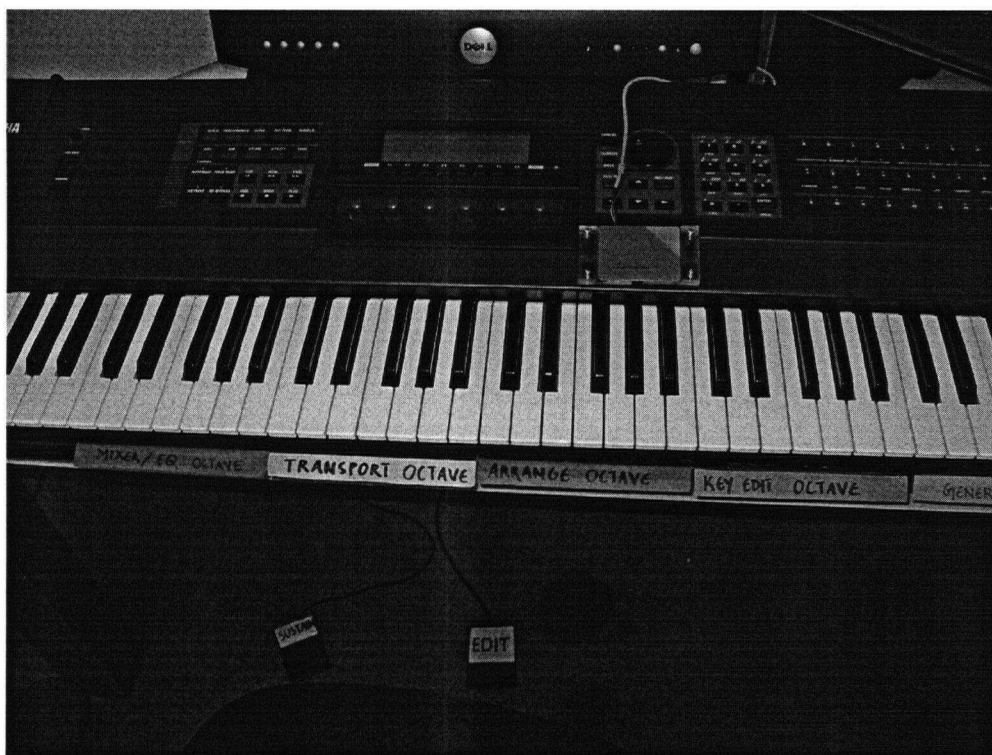


Figure 4.4: Prototype II Piano and Pedal Setup

4.2.4 Analysis of Results

The average learning time among the ten composers practicing the different modes and functions with the *KEYed* user interface prototype II is approximately 16 minutes.

As discussed in the previous section, the dependent variable is the task completion time, and the two independent variables used are the interaction method and the task complexity. Figure 4.5 and Figure 4.6 show the distribution of the task completion times of the composers with different interaction methods and task complexities.

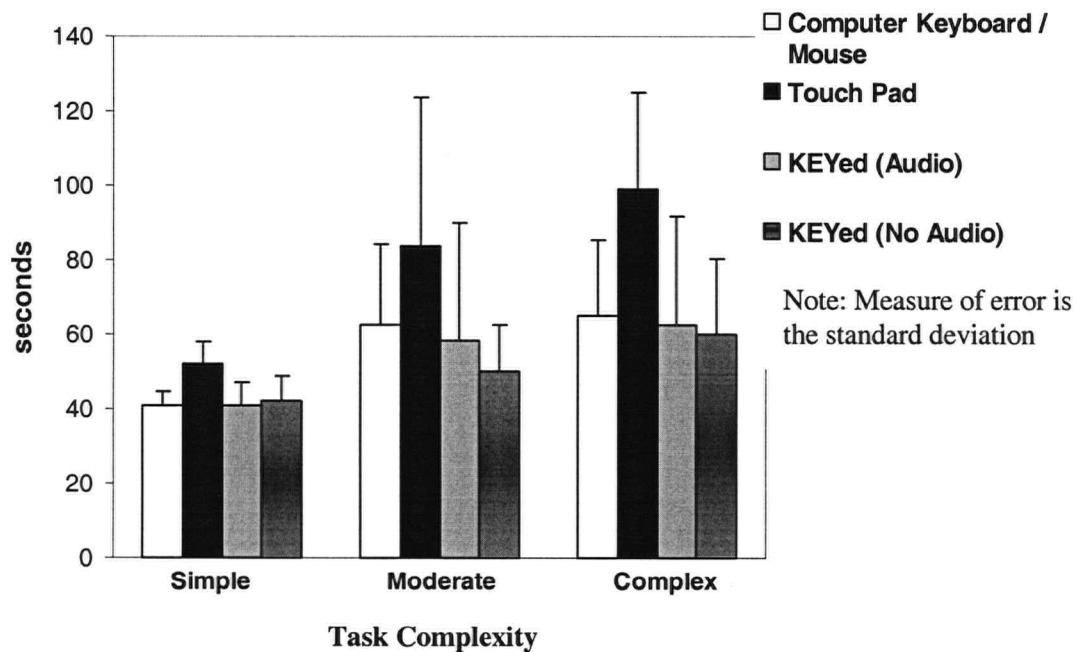


Figure 4.5: Mean Completion Times of the Tasks Complexities for the Different Interaction Methods for 10 Composers

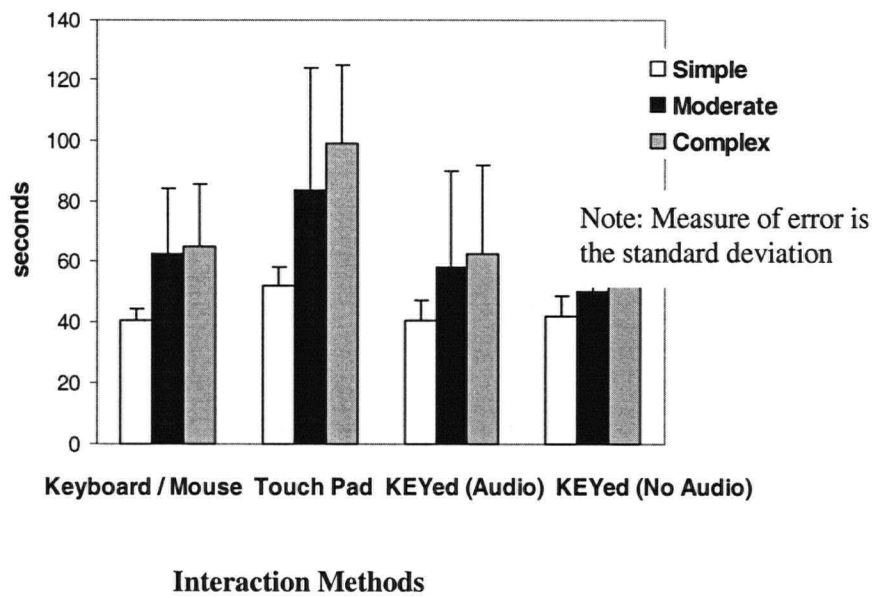


Figure 4.6: Mean Completion Times of the Interaction Methods for the Different Task Complexities for 10 Composers

A two way factorial ANOVA was performed on the reaction times for Task A (Appendix A.6), Task B (Appendix A.7) and Task C (Appendix A.8), with four different interaction methods discussed in Sub-section 4.2.1. Figures 4.7 and 4.8, as shown in the following page, are plotted to visualize any interaction effects between the interaction methods (input device) used and the task complexities.

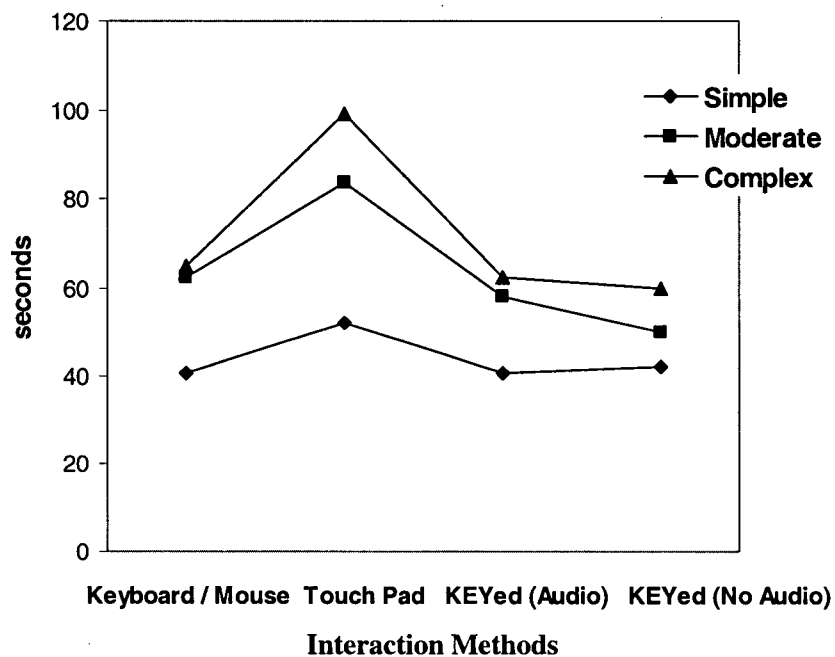


Figure 4.7: Interaction Graph A to Visualize Interaction Effects between the Task Complexities and the Interaction Methods

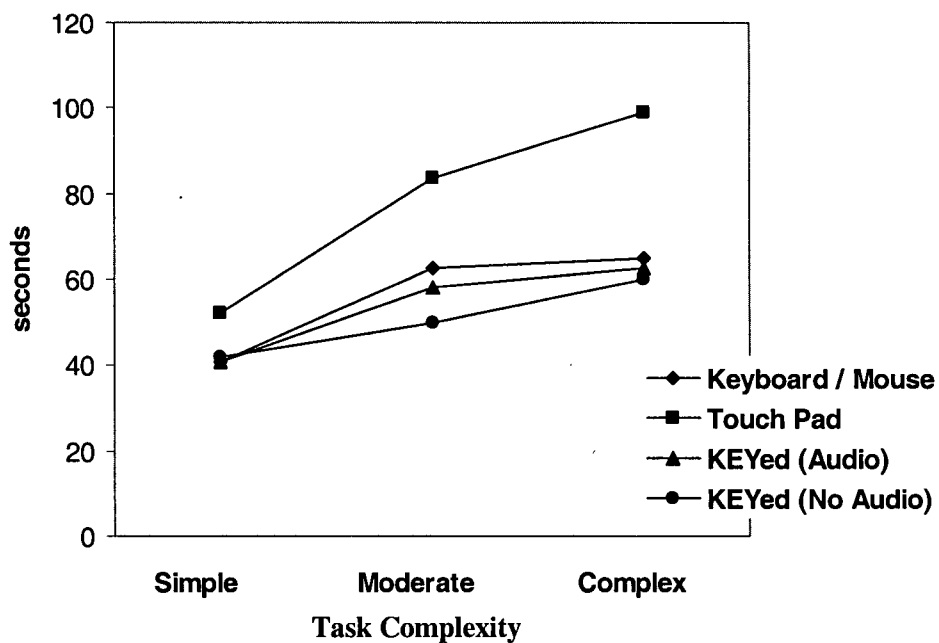


Figure 4.8: Interaction Graph B to Visualize Interaction Effects between the Task Complexities and the Interaction Methods

Table 4.1 is a summary of the ANOVA results on all the interaction methods for complexities.

SOURCE	SS	df	MS	F
Input Device	14238.31	3	4746.1	9.99
Task Complexity	16240.35	2	8120	17.098
Input D * Task C	3037.34	6	506.22	1.066
Within Groups	51288.9	108	474.9	
Total Variability	84804.92	119		

Table 4.1: ANOVA Summary for all the Interactions

The findings from Table 4.1 are as follows:

- 1) For alpha levels of $p \leq 0.05$ and $p \leq 0.01$, a significant main effect of interaction method (input device) is found, $F(3, 100) = 9.99$, $p < 0.05$ & $p < 0.01$.
- 2) For an alpha level of $p \leq 0.05$ and $p \leq 0.01$, a significant main effect of task complexity is found, $F(2, 100) = 17.098$, $p < 0.05$ & $p < 0.01$.
- 3) No significant interaction effect between the interaction method and the task complexity is found.

Table 4.2, shown in the following page is a summary of the ANOVA results between the computer keyboard/mouse interaction method and the KEYed (Audio) interaction method for complexities.

SOURCE	SS	df	MS	F
Input Device	74.75	1	74.75	0.16
Task Complexity	6177.74	2	3088.87	6.65
Input D * Task C	50.51	2	25.26	0.054
Within Groups	25089.84	54	464.62	
Total Variability	31392.9	59		

Table 4.2: ANOVA Summary for Keyboard / Mouse Interaction vs KEYed (Audio)

The findings from Table 4.2 are as follows:

- 1) For alpha levels of $p \leq 0.05$ and $p \leq 0.01$, a significant main effect of task complexity is found, $F(2, 54) = 6.65$, $p < 0.05$ & $p < 0.01$.
- 2) No other significant results are found.

Table 4.3 is a summary of the ANOVA results between the computer keyboard / mouse interaction method and the KEYed (No Audio) interaction method for the task complexities.

SOURCE	SS	df	MS	F
Input Device	439.3	1	439.3	1.73
Task Complexity	4659	2	2329.5	9.179
Input D * Task C	491	2	245.5	0.97
Within Groups	13704.63	54	253.79	
Total Variability	19293.9	59		

Table 4.3: ANOVA Summary for Keyboard / Mouse Interaction vs KEYed (No Audio)

The findings from Table 4.3 are as follows:

- 1) For alpha levels of $p \leq 0.05$ and $p \leq 0.01$, a significant main effect of task complexity is found, $F(2, 54) = 9.179$, $p < 0.05$ & $p < 0.01$.
- 2) No other significant results are found.

Discussions based on these results and the findings from the provided questionnaires are given in the Chapter 5, Section 5.2.

4.2.5 Case Study with Prototype II

Finally, as discussed earlier, a case study is conducted with the *KEYed* user interface prototype II. This study examines the effectiveness of the interface when used during long periods of time in a home project studio.

In this study, the composer is provided with consent form B (Appendix A.10) and asked to use the *KEYed* user interface prototype II as part of his digital music workstation setup and routine composition work. The prototype is installed in his home project studio for a one week period. The total participation time in this study is approximately two hours per day for a total of seven days.

We observe the composer work with this setup several times during this one week period to analyze how the system is used, and how it can be improved for future versions. At the end of the study, the composer is asked to provide feedback on his experience with the system, and further, we have several informal discussions with him. Though we realize that quantitative results from such studies with a single subject are not statistically valid, we conduct a task completion time test before and after the case study for the four different interaction methods, and the task complexities, discussed in Sub-section 4.2.1, to watch for any significant change in patterns when the interface is used for long periods of time.

4.2.6 Case Study Results

Some of the feedback provided by the composer on his experience with the KEYed user interface prototype II follows:

- *Because only a few commands are presently mapped to the MIDI keyboard, one can only perform limited tasks.*
- *I would like to test it with all commands mapped to the MIDI keyboard, and learn to perform complex/realistic tasks without the mouse & computer keyboard.*
- *I wonder what it would be like to map the number pad to a section of the keyboard, for entering numeric values.*
- *I would like the touchpad to be considerably bigger - right now it's too painful to navigate on a fine scale; even a joystick might be better - some keyboards have those.*
- *Right now you've got it set up so that you press, for example, C to open the transport window, and then D, E, etc to perform commands within that window. Would it be possible to perform those commands without having the transport window open? And, there might be other keyboard mappings worth testing. For example, a C major chord (keys pressed approx. simultaneously) could correspond to play, F major to record, c minor to stop?*
- *Overall, an excellent idea - it's about time someone did this.*

Figures 4.9, 4.10, and 4.11 show the distribution of the task completion times of the composer with different interaction methods for the simple, moderate and complex tasks before and after the study.

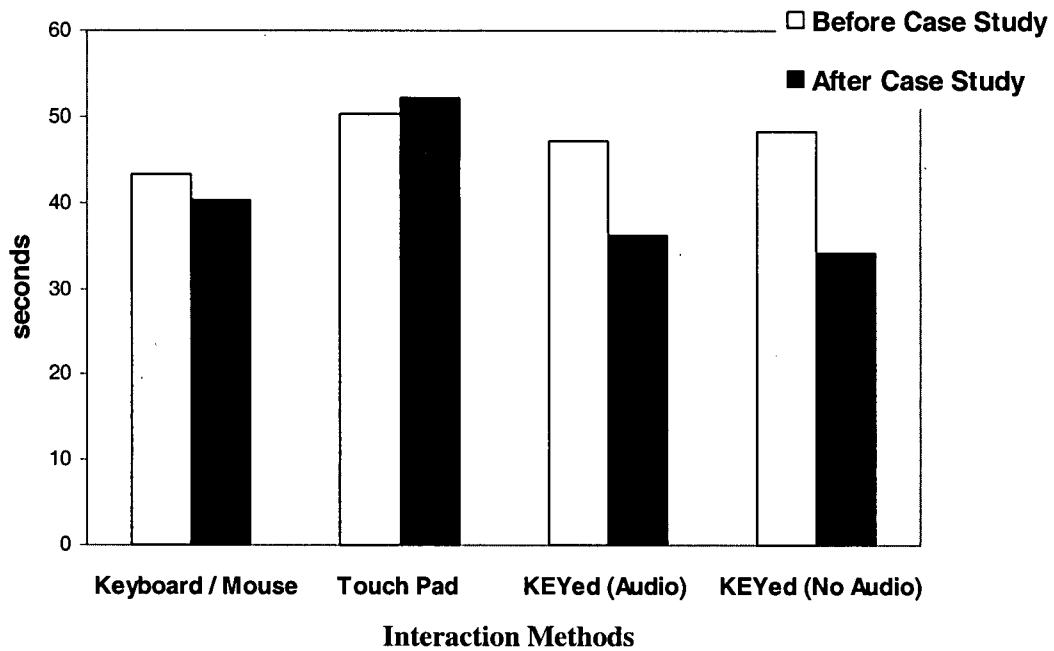


Figure 4.9: Task Completion Times for the Different Interaction Methods for the Simple Task, Before and After the Case Study with One Composer

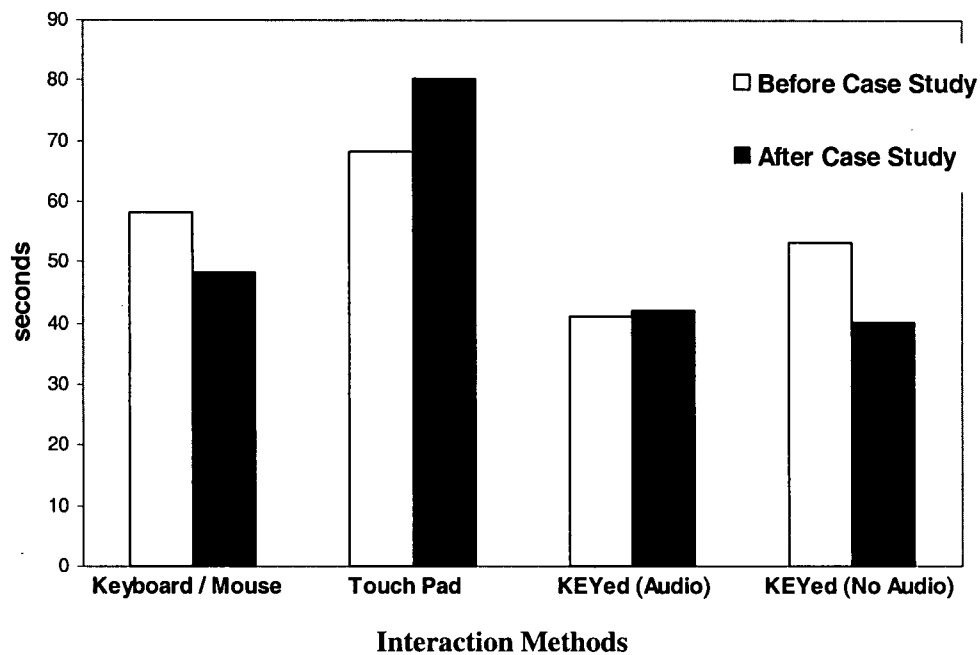


Figure 4.10: Task Completion Times for the Different Interaction Methods for the Moderate Task, Before and After the Case Study with One Composer

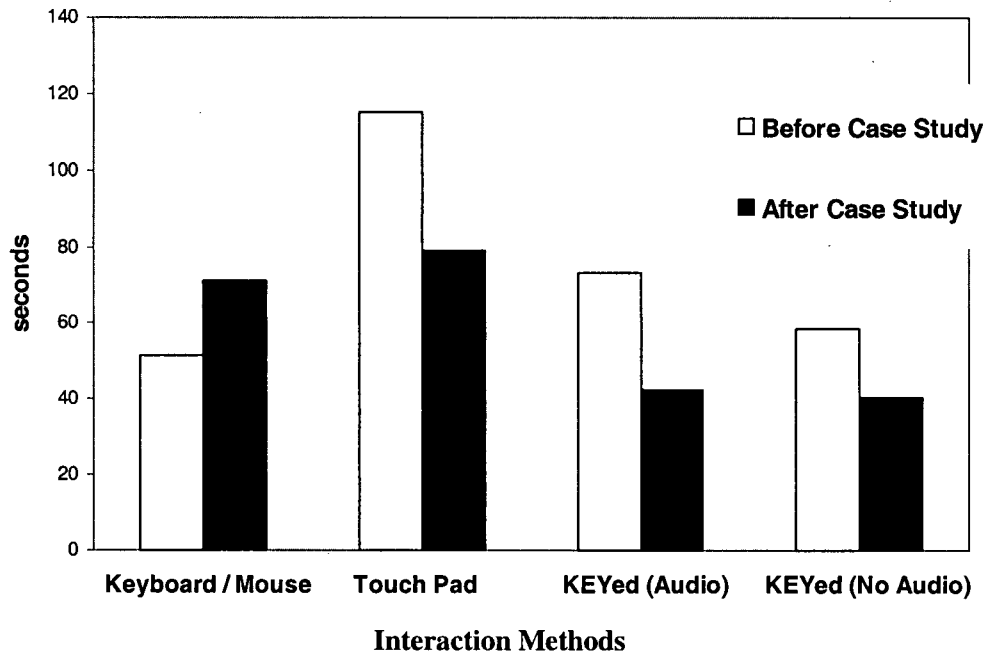


Figure 4.11: Task Completion Times for the Different Interaction Methods for the Complex Task, Before and After the Case Study with One Composer

Discussions based on our findings from the case study are given in Chapter 5, Section 5.3.

4.3 Comparison of the Predicted and Measured Task Completion Times

In this section, we furnish the predicted Keystroke Level Model (KLM) task completion times for the various tasks used in our studies. Comparing such predicted times to the measured task completion time gives insights into the individual operations involved in the tasks. The estimated times shown in Table 2.1 is used for modeling the **K, B, P, H** and **M** Operators. These predicted values and the measured values collected through our user studies are furnished in the below Table 4.4.

		Measured Means (Seconds)	Predicted Time A (Seconds)	Predicted Time B (Seconds)
Task A (Appendix A.2)	Computer Keyboard	116.2218333	67.4 + R1	73.4 + R1
	KEYed Prototype I	101.3736667	58.6 + R1	58.6 + R1
Task B (Appendix A.3)	Computer Keyboard	96.71233333	56.2 + R2	62.2 + R2
	KEYed Prototype I	86.05816667	42.2 + R2	42.2 + R2
Task C (Appendix A.6)	Computer Keyboard (CK) / Mouse (M)	40.74	CK: 44.4 + R3	CK: 46 + R3
			M: 39.8 + R3	M: 41.4 + R3
	KEYed Prototype II (with audio)	40.7399	32.2 + R3	32.2 + R3
	KEYed Prototype II (without audio)	42.1412	32.2 + R3	32.2 + R3
	Touchpad	52.1371	39.8 + R3	41.4 + R3
Task D (Appendix A.7)	Computer Keyboard (CK) / Mouse (M)	62.6374	CK: Not measurable	CK: Not measurable
			M: 41.3 + R4	M: 43.3 + R4
	KEYed Prototype II (with audio)	58.245	22.6 + R4	22.8 + R4
	KEYed Prototype II (without audio)	50.0287	22.6 + R4	22.8 + R4
	Touchpad	83.6633	41.3 + R4	43.3 + R4
Task E (Appendix A.8)	Computer Keyboard (CK) / Mouse (M)	64.923	CK: Not measurable	CK: Not measurable
			M: 44.2 + R5	M: 46.4 + R5
	KEYed Prototype II (with audio)	62.5232	28.8 + R5	29.2 + R5
	KEYed Prototype II (without audio)	59.9132	28.8 + R5	29.2 + R5
	Touchpad	99.1176	44.2 + R5	46.4 + R5

Table 4.4: Measured Mean Task Completion Time vs Predicted Keystroke-Level Model (KLM) Task Completion Time (A, B) where: Predicted Means A is the predicted times with the standard KLM, Predicted Means B is the predicted times with

the refined KLM, and R1, R2, R3, R4, R5 are the task checking time which are further expanded in the following paragraph.

Note that the first set of tasks, Task A and B are the tasks used in the prototype I studies, and the following tasks Task C, D and E correspond to the simple, moderate and complex tasks used in the prototype II respectively. The predicted time A is sum of the actual predicted time using the KLM and the time to read the tasks during the experiments denoted by R1, R2, R3, R4, R5. Predicted time B is the sum of the refined predicted time with a 50% increase in the homing time and the time to read the tasks during the experiments denoted by R1, R2, R3, R4, R5. We assume such an increase in the current homing time (0.4 seconds) used in the Keystroke Level Model due to the music workstation setup as shown in Figure 1.1. Note that certain tasks which involved 1 and 2 degree of freedom (DOF) operations like sliding a fader and drawing a volume graph, were not measurable with the computer keyboard. Discussions based on these comparisons are furnished in the following Chapter, Section 5.4.

Chapter 5

Discussion

This chapter focuses on the results from Chapter 4, in detail. The following sections start by discussing the results from the individual experiments in our study, and then lead to detailed discussions based on these results.

5.1 Preliminary Studies with Prototype I

In our studies with the KEYed user interface prototype I, we see that with an average learning time of approximately 5 minutes, composers perform both tasks significantly better with the KEYed user interface in Test 1 ($p < 0.05$, t-Test). Although Test 2 ($p < 0.06$, t-Test) was done faster than Test 1, as shown in the Figure 4.1, the task completion times, though faster, are not statistically significant with the piano keyboard. We suspect this is due to the minimal learning time, and small number of subjects used for the study.

Subjects gave the computer keyboard a significantly lower rating than the piano keyboard ($p < 0.02$, t-Test) on the questionnaire as shown in the Figure 4.2. They found the piano interaction intuitive and an effective link between playing and editing as shown in Appendix A.11. One composer even said “*Within a year there will be two in every American home.*” Though the no audio condition is not tested in this study, from the questionnaires we learnt that the earcons [6] are effective for performing certain tasks, because the composers already acquired auditory familiarity with the piano keyboard. Composers also suggested that the presence of sounds while editing is a new and interesting concept, although, we suspect the unmodified sounds to be a hindrance, especially because compositions are typically done on particular musical scales.

Interestingly, no mode errors were detected during the tests involving the *KEYed* user interface. We suspect that this is due to the composers' familiarity with the damper and sustain pedals, which are momentary pedals as well, and are commonly used in their industry.

In the early design of this study, we realized that training is an important part of this interface in order to achieve positive performance results. Explicit labeling of the individual keys and the recipe like tasks, we had adopted in this study are not appropriate reflections on how composers work normally. It was important for us to address this issue for our future studies.

Though the results from this study are encouraging, it is necessary for us to perform a detailed study on an improved prototype with more subjects, especially due to the small number of subjects used and the limited functionalities of the interface.

5.2 Controlled Experiments with Prototype II

In this study with the *KEYed* user interface prototype II, we see that with an average learning time of approximately 16 minutes, the composers perform faster with the *KEYed* (Audio) and *KEYed* (No Audio) levels, than the computer keyboard/mouse interaction method; however, these results are not statistically significant, as shown in Figure 4.5, Figure 4.6, Table 4.2 and Table 4.3. Overall, as expected, the mean task completion times significantly increased with task complexity ($F(2, 100) = 17.098, p < 0.05$ & $p < 0.01$). However, with the computer keyboard/mouse and the *KEYed* interaction methods, the difference in the task completion times between the moderate and complex tasks is minimal. The touchpad interaction consistently increases in task completion time with the increase in task complexity. Further, there is no statistically significant interaction effects

between the input device used and the task complexity, as shown in Figure 4.7, Figure 4.8 and Table 4.1.

To reduce the training period with the *KEYed* user interface, the octaves are labeled to show the windows they are mapped to. The tasks presented to the composers are also stated in a more general form, as opposed to the explicit recipe kind of statements used in the experiments with the prototype I.

The subject sample chosen for this study are moderate to expert composers who use digital music workstations extensively. Although all composers are very familiar with standard computer interfaces, such as the keyboard and mouse, and the piano keyboard interface, the pre-test Questionnaire A results show that nine out of the ten composers have very little experience with the touchpad interface. This aligns with our anticipation that the touchpad is a fairly unfamiliar device in music composition workstations. To test for its effectiveness, we had the composers perform all three tasks exclusively with a single-point touchpad as another condition, as opposed to the touchpad used juxtapose with the piano keys as in the *KEYed* user interface prototype II. The composers were earlier trained on the touchpad interfaces' features. As expected, Figure 4.5 and Figure 4.6 show that the single-point touchpad is ineffective in performing the various degrees of tasks. Composers found the touch pad difficult to use, especially for tasks such as pointing and clicking.

Interestingly, after the touch pad is used as part of the *KEYed* user interface prototype II, the results from Questionnaire B (Appendix A.5) for Task B and Task C, show that the composers preferred the touchpad to the standard mouse for tasks such as moving faders and turning knobs as shown in Appendix A.11. They reported that the use of the touchpad for performing fine scale operations with one hand, while holding down the

piano note to open the mixer or EQ windows is more comfortable and natural to use. This also augments the results of the studies where two hands engaged in different motor control mechanisms outperformed the one handed conditions for browsing tasks [27]. It was emphasized earlier in our literature that in humans, the most skilled manual activities involve two hands playing different roles. Guiards' Kinematic Chain model on human bimanual action strongly suggests that the two human hands work in a cooperative and asymmetric manner. Further, studies have also showed that the dominant hand tends to act later, work in a smaller but finer scale and operate within the frame of reference provided by the non-dominant hand [22], which is a common action we observed when the composer used the touchpad in the *KEYed* user interface prototype II.

In this study, we also anticipated that the mean completion times with the *KEYed* user interface (Audio) interaction method would be less than the *KEYed* user interface (No Audio) interaction method. Interestingly, we found out that the *KEYed* user interface (No Audio) level mean completion time is less in moderate and complex tasks, although not statistically significant. Though most composers liked the presence of the earcons [6] emanating from the piano while in the edit mode as shown in Appendix A.11 and as one composer said "*I like the musically arranged editing*", which is consistent with the prototype I studies, they performed slower than expected. We contend that the composers would have performed better with the presence of the earcons, if the tasks were repeated, which is something we did not include as part of this experiment. One composer pointed out that the earcons used in our study, which are basically sounds from the piano keyboard, might confuse a composer while he is composing on the same or different musical scales.

We anticipated this problem before our studies with prototype I. One solution to this problem is to use non musical sounds while in the edit mode.

In the post-test section of Questionnaire B, most composers agreed that the *KEYed* pedal is easy to use. This coincided with the fact that most composers did not create any mode errors both while switching between the composing and editing modes and while switching between the sustain and the *KEYed* pedals, with the exception of one composer. As discussed earlier, this is due to the composers' familiarity with the damper and sustain pedals, which are momentary pedals as well, and are commonly used in their industry. It is interesting to note that the one composer who committed two mode errors did so while switching between the sustain and the *KEYed* pedals. This composer, though familiar with a piano keyboard, is not a regular user of a piano. We contend that such errors in mode switching can be reduced by practice, as shown by pianists who play with their fingers while switching back and forth between the sustain and damper momentary pedals. Further, the composers' comfort and ease in using the pedal coincides with the previous studies that have shown that when an interface chunks various subtasks into one task with appropriate feedback, it performs better [20]. Similarly, the *KEYed* pedal acts as a glue that ties the subtasks; for example, after depressing the pedal, the subtasks *open a window by holding down a piano note, move the scroll bar in the window with the touchpad and release the notes*, gets chunked into a single task by the foot proprioceptive feedback while being engaged with the pedal, which further reinforces the system state and reduces mode ambiguities [19] [1] [2].

In the case of the *KEYed* user interface, apart from the feedback from the earcons (if a non musical sound is used, as suggested) and kinesthetic feedback from the foot used

for reducing mode errors, there is an additional level of kinesthetic feedback and referencing which exists due to the holding of piano notes with certain fingers for opening a window while engaging in tasks within that window with other fingers. For example, for opening a EQ window, a composer depresses the *KEYed* pedal, holds the note 'C3' to open the mixer window, and holds the note 'D3' to open the EQ window within the mixer window while holding the previous note. The positioning of the first and the second finger gives an additional level of kinesthetic feedback, and further, the finger which acts on the second note operates within the frame of reference provided by the first finger, a concept seen earlier in bimanual tasks.

In our discussions in Section 2.2.2, we had seen that previous training on piano performance enables the improvement of the tacit muscle memory required to perform automated music structures and units [15]. A combination of such feedback mechanisms, coupled with the acquired subjective knowledge of the composer, enables the composer to perform tasks easier and faster. For example, in our study, when the composers' objective is to manipulate the master fader, holding the notes 'C3' and 'C#3' enabled him to open the mixer window and select the master fader, as opposed to opening the windows and selecting the fader, typically done using the mouse. Since the design of most of the mappings in the *KEYed* user interface is done in this fashion, composers in our study expressed comfort and stated that they found the process of performing such tasks automatic when using the *KEYed* user interface. An additional reason for such positive responses from the composers is due to the highly spatially multiplexed design of most of the functions in the interface. Each piano key facilitated control of a specific virtual function of the software, which is further laid out in an octave structure where the different

octaves on the piano are mapped to specific windows in the software. Though not included in the *KEYed* user interface, provisions for reconfiguring these mappings help composers create their own custom settings in the future. Although the touch pad is a highly time multiplexed device as a stand alone unit, it performs well when used along with the piano interface.

5.3 Case Study with Prototype II

In the case study with our final prototype for a period of one week, we received interesting feedback from the expert composer about the *KEYed* user interface prototype II, as discussed in Section 4.2.6. Though statistically not valid, Figure 4.9, Figure 4.10 and Figure 4.11 show the task completion times of the composer while using the different interaction methods for the simple, moderate and complex tasks, before and after the case study. The figures show that the *KEYed* (Audio) and *KEYed* (No Audio) interaction methods show a reduction in task completion time when compared to the keyboard/mouse interaction method. In the moderate task, the *KEYed* (Audio) interaction method performed slower after the case study than before the case study. This was a strange case and we were unable to find an answer for such a result after studying the moderate task in detail. However, the composer expressed that he was tired while performing the tasks during our discussions with him after the experiment. We contend that this could have been a reason for such a result. The composer also found the touchpad very hard to use as a stand alone unit. His performance on the experiments reflects this. He suggested the use of a bigger touchpad or a joy stick as a substitute for the current touchpad.

Further, the composer found the piano keyboard mappings to be limiting. He wants more mappings on the piano keyboard, which also implies that he wants to use the interface

more. A solution to this space limitation is to explore further musical structures, such as chords to select more windows and functions within the windows. The other alternative solution is to use the *KEYed* user interface just for the most commonly used macros as hot piano keys. Thereby, this interface would augment the computer keyboard and mouse, although minimizing the use of the latter.

In essence, the composer found the *KEYed* user interface prototype II to have affordances for the faster recollection of commands to perform operations, to be more comfortable and natural to use, and to take less space in his studio. He found the *KEYed* foot pedal easy to acquire and an ideal mode switch for switching between the performing and editing modes. He preferred the *KEYed* interaction to the computer keyboard/mouse interaction.

5.4 Comparison of the Predicted and Measured Task Completion Times

This section discusses about the differences between the predicted task completion times based on the Keystroke Level Model and the measured task completion times, for the different tasks used in our experiments as shown in Table 4.4. Some of our important observations and inferences are as follows:

a) Firstly, when we compare the predicted versus measured task completion time we see that the measured task completion time is significantly high in all the tasks. This is primarily due to the time to read the tasks during the experiments denoted by R1, R2, R3, R4, R5, which we add to the predicted times for a more accurate comparison. R1 and R2 are anticipated to be longer as the task descriptions Task A and Task B are in point form as shown in Appendix A.3 and A.4. R3, R4, R5 are anticipated to be shorter as the task

descriptions Task C, Task D and Task E are in a general form as shown in Appendix A.6, A.7 and A.8.

b) In almost all the tasks we see that the difference between the measured task completion time between the computer keyboard /mouse or computer keyboard conditions and the KEYed user interface conditions, is a lot less compared to the difference between the predicted task completion time between the computer keyboard /mouse or computer keyboard conditions and the KEYed user interface conditions. We anticipate such a result due to the lesser training time given to the composers with the KEYed user interface. We contend that the **M** operator is more in this case due to the reduced training period and thereby trying to remember the piano keyboard mappings to perform the required task.

c) In Task C, we assume the predicted time for the computer keyboard / mouse combination to be an average of the predicted individual computer keyboard and the mouse conditions. In this task we notice that the effect of R3 is more due to the more generalized form of the task descriptions and validated by the smaller difference between the measured and predicted mean times for the KEYed user interface conditions. However, R4 and R5 seem to have a lesser effect, validated by the bigger difference between the measured and predicted mean times for the KEYed user interface conditions in spite of the generalized form of the task descriptions. As these tasks (Task D and Task E), requires the use of the touch pad interface we contend that this effect has to with the composers' interaction with the touchpad. Though the composers expressed comfort in using the touchpad we notice an increase in the measured task completion time involving the device.

Conclusion and Future Work

This chapter summarizes the goals, results and contributions of this thesis, and outlines some directions for further work in this research area.

6.1 Overview and Conclusion

Our motivation originates from observing the awkwardness of the existing user interfaces in music composition workstations. User interfaces in music workstations have become cumbersome, especially as they require the use of multiple input devices, such as an electronic piano keyboard, a computer keyboard, and a mouse used repetitively during a composing task. Considering this, our goal is to give the composer a more transparent [16] interface which allows him to focus on the creative aspects of music composition.

Early in our studies, we saw how significant user interface research exists in the field of music synthesis and performance, as opposed to music composition workstations, which researchers consider as more of an off-line editing tool. However, considering the modern music production practices, we found that the principles and findings in the field of Human Computer Interaction (HCI) can be applied to the design of novel composer computer interfaces. New interfaces for performing such music production functions may greatly improve the ease of use of such production workstations, and enhance the overall musical expressive abilities of the composer.

Earlier, we observed that it is possible to reduce the number of input devices used in current music production workstations, by moving some of the more commonly used computer keyboard macros and the mouse functions to the master controller (usually an

electronic piano keyboard). This would allow the composer to work more efficiently, because all common functions could then be accessed using only the master controller. This observation gave birth to the *KEYed* user interface, in which the mappings of the music production functions are well laid out in the electronic piano keyboard itself. The customizable piano mapping provides the composer with a familiar configuration of space and sound, allowing him or her to focus on the creative aspects of music composition. Features for complex sound editing and control are integrated into the system; therefore the user interface requires far fewer operations for achieving various production tasks. This helps the composers focus on musical rather than operational issues.

The results from our experiments with the *KEYed* user interface show that, though the composers were not able to perform tasks significantly faster, especially with our prototype II interface, when compared to conventional methods, they experienced enormous comfort, naturalness and intimacy when engaged with the new interface. Further, the interface did not take additional space in the studio, as opposed to the alternative input devices discussed in Section 2.1. These results from our studies with the *KEYed* user interface prototypes also validate our approach to the design, which shows that the results from human computer interaction literature may be used as tools for developing methodologies for the design and evaluation of composer computer interfaces. Our studies with the *KEYed* user interface validates Guiards' Kinematic Chain Model which states that the dominant hand tends to act later, work in a smaller but finer scale and operate within the frame of reference provided by the non-dominant hand [22], which is a common action we observed when the composer used the touchpad in the *KEYed* user interface prototype II. Further, we learnt that the Kinematic Chain Model can be extended for tri channel

operations as in the case of the *KEYed* user interface where a lower bandwidth channel such as the foot which acts earlier, sets the reference for the manual operations. The results from our studies suggest that the two human hands and a foot, work in a cooperative and asymmetric manner.

The conclusion underlying this thesis is that user interfaces can be made more usable by careful attention to detail at the design stage. This is only possible if the designer can think about the interfaces from the users' point of view by carefully considering her domain and acquired subjective knowledge, and by providing mechanisms for adequate feedback through appropriate channels. Applying such a user centered approach requires that the product be prototyped at the earliest stages to identify the users' strategies and expectations. Our results from the experiments conducted with the *KEYed* user interface prototypes show that applying such design methodologies facilitates the design of novel and usable interfaces for music composition workstations.

6.2 Future Work

Some of the future directions for work in this research area are as follows:

1. Exploring musical structures for improving mappings.
2. Applying the concepts of the *KEYed* user interface to other musical instruments used as music composition master controllers.
3. Exploring alternative techniques for multi-degree of freedom tasks in composition workstations.
4. Providing user customizable mappings.

6.3 Contributions

Some of the key contributions of this thesis are as follows:

1. We have shown that apart from facilitating music performance, a piano keyboard can be used as an alternative input device in music composition workstations. By using the octave structure of the piano and a key based segmentation, the piano keyboard can facilitate mappings and the acquisition of software windows and functions within the windows.
2. We found that a momentary foot pedal like the *KEYed* foot pedal is easy to acquire, and is an ideal mode switch for switching between the performing and editing modes. Further, these pedals are effective and easy to use even when two momentary foot pedals are used for mode switching and require switching back and forth between the pedals.
3. We have shown that a single point touchpad is effective for bimanual tasks. For example, the use of the touchpad for performing fine scale operations with one hand, while holding down the piano note to open the mixer or EQ windows is found to be natural to use. Composers found the touchpad hard to use when used exclusively for pointing and selecting tasks.
4. In general, the *KEYed* user interface illustrates how an appropriate mapping of the layout, feedback, and context is important in the design of user interfaces.
5. Further, we have disseminated our results at peer reviewed conferences such as CHI'2002 [46] and ICMC'2002 [47].

Bibliography

- [1] Sellen, A., Kurtenbach, G. and Buxton, W. "The Prevention of Mode Errors Through Sensory Feedback." *Human-Computer Interaction*, 1992 v.7 n.2, pp. 141-164.
- [2] Sellen, A., Kurtenbach, G. and Buxton, W. "The Role of Visual and Kinesthetic Feedback in the Prevention of Mode Errors Interactive Technologies and Techniques." *Proceedings of Human-Computer Interaction*, 1990, pp. 667-673.
- [3] Zhai, S., Smith, B., and Selker, T. "Dual Stream Input for Pointing and Scrolling." *Proceedings of ACM Conference on Human Factors in Computing Systems*, 1997 v.2, pp. 305-306.
- [4] Anderson, T., Smith, C. "Composability: widening participation in music making for people with disabilities via music software and controller solutions", *Proceedings of the ACM conference on Assistive technologies*, April 11 - 12, 1996, Vancouver, BC.
- [5] Clarke, M. "Composing with multi-channel spatialization as an aspect of synthesis", *Proceedings of International Computer Music Conference*, 1999, pp. 17-19.
- [6] Mynatt, E. "Auditory Presentation of Graphical User Interfaces." *Auditory Display*, Ed. Gregory Kramer, SFI Studies in the Sciences of Complexity, Proc. Vol. XVIII, Addison-Wesley, 1994.
- [7] Gaver, W., and Smith, R. (1990). "Auditory Icons in Large-Scale Collaborative Environments." In D. Diaper *et al.* (Eds.) *Human Computer Interaction - INTERACT'90*, Elsevier Science Publishers B.V. (North Holland), pp. 735-740.

- [8] Brewster, S.A., and Wright, P.C. & Edwards, A.D.N. (1993). "An Evaluation of Earcons for Use in Auditory Human-Computer Interfaces." *Human Factors in Computing Systems, INTERCHI '93*, IOS Press (Amsterdam), pp. 222-227.
- [9] Mereu, S.W., and Kazman, R. (1996). "Audio Enhanced 3D Interfaces for Visually Impaired users." *Human Factors in Computing Systems. Common Ground. CHI '96 Conference Proceedings*, ACM (New York), pp. 72-78.
- [10] Norman, D. A., "Cognitive Engineering", **In:** Norman, D. A. and Draper, S. W. *User centered system design*, Lawrence Erlbaum Associates, Inc., 1986, pp. 31-61.
- [11] Preece, J., **In:** *A guide to usability, human factors in computing*, Addison-Wesley Publishing Company, 1993.
- [12] Nielsen, J., **In:** *Usability Engineering*, Academic Press, Inc.
- [13] Carroll, J. M., Mack, R. L. and Kellogg, W. A. "Interface metaphors and user interface design." *Handbook of human-computer interaction*, Helander, M. (ed)
- [14] Weiser, M. "The world is not a desktop", *Interactions*, vol. 1 no. 1, 1994, pp. 7-8.
- [15] Pamela, S "Acquiring a Dancer's Intuition", May 2002.
http://www.gse.harvard.edu/~t656_web/Spring_2002_students/sparkman_pamela_acquiring_dancers_intuition.html
- [16] Fels, S. "Intimacy and Embodiment: Implications for Art and Technology." *Proceeding of the ACM conference on Multimedia*, pp. 13-16, 2000.
- [17] Lewis, C. and Norman, D. "Designing for error", *Human Computer Interaction, Toward the year 2000*, pp. 686-697.

- [18] Norman, D.A. (1981). "Categorization of action slips." *Psychology Review*, 88(1), pp. 1-15.
- [19] Monk, A. (1986). "Mode Errors: A user-centred analysis and some preventive measures using keying contingent sound." *International Journal of Man-Machine Studies*, 24, pp. 313-327.
- [20] Buxton, W. (1986). "Chunking and Phrasing and the design of Human-Computer Dialogues." *Proceedings of the IFIP World Computer Congress*, Dublin, Ireland, pp. 475-480.
- [21] Fitzmaurice, G. and Buxton, W. (1997). "An empirical Evaluation of Graspable User Interfaces: towards specialized, space-multiplexed input." *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI'97)*, pp. 43-50.
- [22] Guiard, Yves (1987). "Asymmetric Division of Labor in Human Skilled Bimanual Action: The Kinematic Chain as a Model." *Journal of Motor Behavior*, 1987, 19, pp. 486-517.
- [23] Leganchuk, A., Zhai, S. & Buxton, W. (1998). "Manual and Cognitive Benefits of Two-Handed Input: An Experimental Study." *Transactions on Human-Computer Interaction*, 5(4), pp 326-359.
- [24] Fitts, P.M. (1954). "The information capacity of the human motor system in controlling the amplitude of movement." *Journal of experimental Psychology* 47, pp 381-391.
- [25] MacKenzie, I. S. (1995). "Movement time prediction in human-computer interfaces." In R. M. Baecker, W. A. S. Buxton, J. Grudin, & S. Greenberg (Eds.), *Readings in*

human-computer interaction (2nd ed.), (pp. 483-493). Los Altos, CA: Kaufmann.
[reprint of MacKenzie, 1992].

[26] Buxton, W. "A Directory of Sources for Input Technologies"
<http://www.billbuxton.com/InputSources.html>

[27] Zhai, S., Smith, B.A., Selker, T. "Improving Browsing Performance: A study of four input devices for scrolling and pointing tasks." *INTERACT 1997*, pp 286-293.

[28] Zhai, S., Milgram, P. William Buxton: "The Influence of Muscle Groups on Performance of Multiple Degree-of-Freedom Input." *CHI 1996*, pp 308-315.

[29] Langolf, G. D., Chaffin, D. B., and Foulke, J. A. "An investigation of Fitts' law using a wide range of movement amplitudes." *Journal of Motor Behavior*, 8, 2 (1976), pp 113-128.

[30] Card, S.K., Mackinlay, J.D., and Robertson, G.G. "The Design Space of Input Devices," *CHI 1990*, pp. 117-124.

[31] Heckl, P. "The Elements of Friendly Software Design," 1991, The New Edition, Sybex.

[32] Ikujiro, N., Hirotaka, T. "The Knowledge Creating Company: How Japanese Companies Create the Dynasties of Innovation," 1995, New York: Oxford University Press.

[33] Jones, P.H. "Tacit knowledge: Information overload and perceived information value", *Computer-Supported Cooperative Work (CSCW 96) Conference*, November 16, 1996, Boston.

- [34] Gould, J., Boies, S., and Lewis, C. "Making Usable, Useful, Productivity Enhancing Computer Applications." *Communications of the ACM*, 34 (1), pp. 74-85.
- [35] Cook, P. "Principles for Designing Computer Music Controllers," *ACM CHI Workshop in New Interfaces for Musical Expression (NIME)*, Seattle, April, 2001.
- [36] Chu, L., "Using Haptics for Digital Audio Navigation," *Proceedings of International Computer Music Conference 2002*, ICMA, pp 118-121.
- [37] Wisneski, C., Hammond, E. "Multi-Parameter Controllers for Audio Mixing," *Proceedings of CHI' 98*, pp. 299-300.
- [38] Card, S.K., Moran, T.P., Newell, A. (1983). "The Psychology of Human-Computer Interaction." Hillsdale, New Jersey: Erlbaum.
- [39] Card, S.K., Moran, T.P., Newell, A. (1980a). "The keystroke-level model for user performance time with interactive systems." *Communications of the ACM*, 23(7), pp. 396-410.
- [40] Joy, W., Horton, M. (1986). "An introduction to editing with vi." In *UNIX User's Supplementary Documents (USD)*. Berkeley, CA: Computer Systems Research Group, University of California, USD.
- [41] Kieras, D. (1993). "Using the Keystroke-Level Model to Estimate Execution Times." <http://www.pitt.edu/~cmlewis/KSM.pdf>.
- [42] Gill, H. J. **In:** *The Tacit Mode: Michael Polanyi's Postmodern Philosophy*, State University of New York Press, 2000.
- [43] Ryle, G. **In:** *The Concept of Mind*. University of Chicago Press, Chicago, USA, 1949.

- [44] Raskin, J. **In:** *The Humane Interface*. ACM Press, New York, NY, USA, 2000.
- [45] Oviatt, S. and Cohen, P.R. (1991). "The contributing influence of speech and interaction on human discourse patterns." **In:** J.W. Sullivan and S.W. Tyler (eds.), *Intelligent User Interfaces*, ACM Press Frontier Series. New York: Addison-Wesley, pp. 69-83.
- [46] Mohamed, F., Fels, S. "LMNKui: Overlaying computer controls on a piano keyboard." *Proceedings ACM Conference on Computer Human Interaction (CHI'02)*, pp. 638-639.
- [47] Mohamed, F., Fels, S. "KEYed user interface: Tools for expressive music production." *Proceedings of the International Computer Music Conference (ICMC'02)* pp. 88-91.

Appendix A

KEYed User Interface Experiments

A.1 Participant Questionnaire for Prototype I

THE UNIVERSITY OF BRITISH COLUMBIA



December 1, 2001

Human Communication Technologies Lab

Department of Electrical & Computer Engineering
2356 Main Mall, Room 155A
Vancouver, BC, Canada,
V6T 1Z4
Phone: 604-822-4583

Post-Test Questionnaire Prototype I

Thank you for participation.

Please take a moment to rate the system in terms of its controls and ease of use. Please circle your answers. You can skip any questions that you do not wish to answer or that do not apply to you.

1. How frequently do you use a software based music composition system?

- 0 - Never
- 1 - Every few months.
- 2 - About once a month.
- 3 - About once per week.
- 4 - All the time, can't get enough of it

2. Are you a piano player?

YES NO

3. Based on your experience how would you rate the following:

	Terrible	Great
QWERTY KEYBOARD ALONE INTERACTION	0 1 2 3 4 5 6	
KEYed USER INTERACTION	0 1 2 3 4 5 6	
THE MIDI FOOT SWITCH FOR MODE CONTROL	0 1 2 3 4 5 6	
THE AUDITORY FEEDBACK FROM THE SYNTH DURING LMNK CONTROL	0 1 2 3 4 5 6	

4. Were you frequently trying to remember how to use the controls? YES NO

5. Did you find the KEYed UI controls too limiting? YES NO

Comments: _____

A.2 Task A used for Prototype I

THE UNIVERSITY OF BRITISH COLUMBIA



December 1, 2001

Human Communication Technologies Lab

Department of Electrical & Computer Engineering
2356 Main Mall, Room 155A
Vancouver, BC, Canada,
V6T 1Z4
Phone: 604-822-4583

TASK A

1. Click 'Start' on the LMNKUT form.
2. Click on the Cubase program.
3. Open the 'Transport window'.
4. Turn the 'click' on, on the transport window.
5. Set the Left Locator to 0001.01.000
6. Turn the 'Punch-In' on, on the transport window.
7. Set the Right Locator to 0020.01.000
8. Turn the 'Punch-Out' on, on the transport window.
9. Go down to the empty 'Track 4'.
10. Press 'Record' on the transport window.
11. Keep sequencing until the record turns off at bar 20.
12. Click 'Stop' on the transport window.

13. Click 'Stop' again to come back to the beginning.
14. Open the 'Key Edit' window on the newly recorded track.
15. Look for the notes you just recorded.
16. Close the 'Key Edit' window.
17. Click on the LMNKUT window on the bar below and click 'Stop'.

A.3 Task B used for Prototype I

THE UNIVERSITY OF BRITISH COLUMBIA



December 1, 2001

Human Communication Technologies Lab

Department of Electrical & Computer Engineering

2356 Main Mall, Room 155A

Vancouver, BC, Canada,

V6T 1Z4

Phone: 604-822-4583

TASK B

1. Turn the 'Click off' on the transport window.
2. Set the Left Locator to 0001.01.000
3. Set the Right Locator to 0005.01.000
4. Go down to the empty 'Track 7'.
5. Press 'record' on the transport window.
6. Keep sequencing until the record turns off at bar 5.
7. Click 'Stop' on the transport window.
8. Click 'Stop' again to come back to the beginning.
9. 'Copy' the track.
10. Do 'two - adjacent' 'Pastes' of the track.
11. 'Cut' the last section of the track by performing a 'Cut'.
12. Click 'Stop' on the transport window.
13. 'Solo' the 'Track 7'.

14. Play the remaining two sections of the 'Track 7' on the transport window for your listening.
15. Click 'Stop' on the transport window at the end of the 'Track 7'.
16. Click 'Stop' again to come back to the beginning.
17. Click on the LMNKUT window on the bar below and click 'Stop'.

A.4 Questionnaire A for Prototype II

THE UNIVERSITY OF BRITISH COLUMBIA



November 6, 2002

Human Communication Technologies Lab

Department of Electrical & Computer Engineering
2356 Main Mall, Room 155A

Vancouver, BC, Canada,

V6T 1Z4

Phone: 604-822-4583

Questionnaire A

Study: Measuring reaction time of three music editing tasks using

- a) A computer keyboard and mouse combination
- b) A 76 key piano keyboard with a foot pedal
- c) A single point touchpad
- d) A combination of piano keyboard with a foot pedal and a touchpad with the audio turned off
- e) A combination of piano keyboard with a foot pedal and a touchpad with the audio turned on

Project Principal Investigator: Dr. Sidney Fels, Department of Electrical and Computer Engineering, 604-822-5338

Co-Investigator: Mr. Farhan Mohamed, M.A.Sc. candidate, Department of Electrical and Computer Engineering, 604-822-4583. This study is conducted by Mr. Farhan Mohamed as part of his M.A.Sc. thesis in the Department of Electrical and Computer Engineering under the supervision of Dr. Sidney Fels.

Purpose

The purpose of this questionnaire is to collect the demographic information that will help the investigators understand the relationship between you and the responses you make during the study.

All responses, including those on this questionnaire will be recorded. Your identity will be confidential and will be known only to the investigators. Do not write your name on this questionnaire. In any publication that arises from this study you will be identified by 3 digit random numbers.

For investigators use only:

Date of session (yyyy-mm-dd):

Subject Number:

THE FOLLOWING QUESTIONS ARE TO BE ANSWERED BEFORE THE EXPERIMENT

NOTE: Fill in the blanks with a word or a "tick" mark, or circle "YES" or "NO".

- (1) Which hand do you normally write with? _____
- (2) Have you used a computer before? YES _____ NO _____

If NO, go to question (3)
If YES, continue to (2)(a)

- (2)(a) How often do you use a computer?

Everyday _____
Every week _____
Bi-weekly _____
Every month _____
Rarely _____

- (2)(b) Have you used a computer keyboard to input words or commands on a computer before (e.g. as an "input device")?
YES _____ NO _____

If NO, go to (2)(d) below
If YES, continue

- (2)(c) How often do you use a computer keyboard as an input device for a computer?

Everyday _____
Every week _____
Bi-weekly _____
Every month _____
Rarely _____

- (2)(d) Have you used a Mouse as an input device for a computer?
YES _____ NO _____

If NO, go to (2)(f)
If YES, continue

- (2)(e) How often do you use a Mouse as an input device for a computer?

Everyday _____
Every week _____
Bi-weekly _____
Every month _____

Rarely _____
(2)(f) Have you used a Touch Pad as an input device for a computer?
YES_____ NO_____

If NO, go to question (3)
If YES, continue

(2)(g) How often do you use a Touch Pad as an input device for a computer?

Everyday _____
Every week _____
Bi-weekly _____
Every month _____
Rarely _____

(3) Do you play a keyboard musical instrument?
YES_____ NO_____

If NO, go to question (4)
If YES, continue

(3)(a) How often do you play a keyboard musical instrument?

Everyday _____
Every week _____
Bi-weekly _____
Every month _____
Rarely _____

(3)(b) Do you play with two hands? YES_____ NO_____

(3)(c) Have you used sustain / damper foot pedals while you play?
YES_____ NO_____

(4) Do you compose music? YES_____ NO_____

If NO, go to question (5)
If YES, go to (4)(a)

(4)(a) Do you use computer based music sequencing software for composing?
YES_____ NO_____

If NO, go to question (5)
If YES, continue

(4)(b) Which music sequencing software do you use for composing?

Logic Audio	_____
Cubase VST	_____
Nuendo	_____
Cakewalk	_____
Protools	_____
Performer	_____
Others	_____
(Specify)	

(4)(c) How often do you use music sequencing software for composing?

Everyday	_____
Every week	_____
Bi-weekly	_____
Every month	_____
Rarely	_____

(4)(d) From the following, select a musical instrument you use as a master controller while using music sequencing software:

Violin	_____
Trumpet	_____
Piano	_____
Flute	_____
Guitar	_____
Others	_____
(Specify)	

(5) Do you currently have, or have you been diagnosed with, or been treated for, any of the following?

- ☐ Hearing difficulties
- ☐ Difficulties seeing the computer screen
- ☐ Difficulties using a foot pedal
- ☐ Difficulties using a computer mouse
- ☐ Difficulties using a touchpad

Add any further comments below:

A.5 Questionnaire B for Prototype II

THE UNIVERSITY OF BRITISH COLUMBIA



November 6, 2002

Human Communication Technologies Lab

Department of Electrical & Computer Engineering
2356 Main Mall, Room 155A
Vancouver, BC, Canada,
V6T 1Z4
Phone: 604-822-4583

Questionnaire B

Study: Measuring reaction time of three music editing tasks using

- a) A computer keyboard and mouse combination
- b) A 76 key piano keyboard with a foot pedal
- c) A single point touchpad
- d) A combination of piano keyboard with a foot pedal and a touchpad with the audio turned off
- e) A combination of piano keyboard with a foot pedal and a touchpad with the audio turned on

Project Principal Investigator: Dr. Sidney Fels, Department of Electrical and Computer Engineering, 604-822-5338

Co-Investigator: Mr. Farhan Mohamed, M.A.Sc. candidate, Department of Electrical and Computer Engineering, 604-822-4583. This study is conducted by Mr. Farhan Mohamed as part of his M.A.Sc. thesis in the Department of Electrical and Computer Engineering under the supervision of Dr. Sidney Fels.

Purpose

The purpose of this questionnaire is to collect your impressions of the experimental tasks and your performance.

All responses, including those on this questionnaire will be recorded. Your identity will be confidential and will be known only to the investigators. Do not write your name on this questionnaire. In any publication that arises from this study you will be identified by 3 digit random numbers.

For investigators use only:

Subject Number:

THE FOLLOWING QUESTIONS ARE TO BE ANSWERED AT THE END OF TASK A

NOTE: Circle or Tick the answer that most closely agrees with you.

- (1) How often did you have difficulties remembering the piano keyboard mappings while editing?

Never Rarely Frequently All the time

- (2) How often did you get the Sustain Pedal and Edit Pedal mixed up while recording and editing?

Never Rarely Frequently All the time

Select your response to the following statement:

- (3) I prefer the sound from the piano keyboard to be turned off while editing.

Strongly Agree Agree Neutral Disagree Strongly Disagree

Add any further comments below:

THE FOLLOWING QUESTIONS ARE TO BE ANSWERED AT THE END OF TASK B

NOTE: Circle or Tick the answer that most closely agrees with you.

Select your responses to the following 4 statements:

- (1) It was easy to select the master volume fader.

Strongly Agree Agree Neutral Disagree Strongly Disagree

- (2) I like the mouse better than the touchpad for sliding the master volume fader.

Strongly Agree Agree Neutral Disagree Strongly Disagree

- (3) I like the location of the touchpad on the piano keyboard.

Strongly Agree Agree Neutral Disagree Strongly Disagree

- (4) I was able to copy and paste MIDI parts faster with the piano keyboard than the computer keyboard.

Strongly Agree Agree Neutral Disagree Strongly Disagree

Add any further comments below:

THE FOLLOWING QUESTIONS ARE TO BE ANSWERED AT THE END OF TASK C

NOTE: Circle or Tick the answer that most closely agrees with you.

Select your responses to the following 3 statements:

- (1) For drawing a volume graph, I prefer the touchpad / piano keyboard combination to the mouse.

Strongly Agree Agree Neutral Disagree Strongly Disagree

- (2) It was easy to turn EQ knobs with the touchpad.

Strongly Agree Agree Neutral Disagree Strongly Disagree

- (3) I like the sound from the piano when holding multiple notes during music editing.

Strongly Agree Agree Neutral Disagree Strongly Disagree

Add any further comments below:

THE FOLLOWING QUESTIONS ARE TO BE ANSWERED AT THE END OF THE EXPERIMENT

NOTE: Circle or Tick the answer that most closely agrees with you.

Select your responses to the following 3 statements:

- (1) The piano keyboard / touch pad combination is easier to use than the computer keyboard / mouse combination for music editing tasks.

Strongly Agree Agree Neutral Disagree Strongly Disagree

- (2) With practice over time, I will probably prefer editing music from the piano keyboard with the touch pad and edit pedal, to editing with the computer keyboard and mouse.

Strongly Agree Agree Neutral Disagree Strongly Disagree

- (3) I like the Edit Pedal for switching between music playing and music editing modes.

Strongly Agree Agree Neutral Disagree Strongly Disagree

Please feel free to skip any of the following questions if you don't have any comments.

- (4) What did you like about editing music with the piano keyboard / edit pedal / touch pad combination?

- (5) What didn't you like about editing music with the piano keyboard / edit pedal / touch pad combination?

(6) Add any suggestions you may have on improving the current piano keyboard / edit pedal / touch pad music editing system:

Add any further comments below:

[illegible]

112

A.6 Task A used for Prototype II

THE UNIVERSITY OF BRITISH COLUMBIA



Human Communication Technologies Lab

Department of Electrical & Computer Engineering
2356 Main Mall, Room 155A

Vancouver, BC, Canada,

V6T 1Z4

Phone: 604-822-4583

TASK A

1. Open the Transport window
2. On the current MIDI track, perform a recording by playing on the piano for the pre-set number of bars with the sustain pedal pressed continuously during the recording
3. After the recording stops, stop playing and move the play cursor to the 0 position
4. Play the bars 0 to 10 of the pre-recorded piece once
5. Stop playing and move the play cursor to the 0 position
6. Close the Transport window

A.7 Task B used for Prototype II

THE UNIVERSITY OF BRITISH COLUMBIA



Human Communication Technologies Lab

Department of Electrical & Computer Engineering
2356 Main Mall, Room 155A

Vancouver, BC, Canada,

V6T 1Z4

Phone: 604-822-4583

TASK B

1. Open the Transport window
2. On the current MIDI track, perform a recording by playing on the piano for the pre-set number of bars with the sustain pedal pressed continuously during the recording
3. After the recording stops, stop playing and move the play cursor to the 0 position
4. Close the Transport window
5. Copy the part you just recorded
6. Forward to bar 20
7. Paste the selected track at bar 20
8. Open the VST mixer
9. Move the master fader from the position 0 to position -30.6, or from -30.6 to 0 on the VST mixer
10. Close the VST mixer

A.8 Task C used for Prototype II

THE UNIVERSITY OF BRITISH COLUMBIA



Human Communication Technologies Lab

Department of Electrical & Computer Engineering
2356 Main Mall, Room 155A
Vancouver, BC, Canada,
V6T 1Z4
Phone: 604-822-4583

TASK C

1. Select MIDI track 2
2. Select the MIDI part on track 2
3. Open the Key Edit window
4. Draw a note for 3 bars
5. Draw a fade-in volume graph on the volume graph area for 2 bars
6. Close the Key edit window
7. Open the Transport window
8. Perform a Recording by playing on the piano for the preset bars with the sustain pedal pressed continuously during the recording
9. After the recording stops, stop playing and move the play cursor to the 0 position
10. Close the Transport window

11. Open the EQ window by opening the mixer
12. Turn the Hi-Mid Gain knob to +24
13. Close the EQ and Mixer windows
14. Open the EQ window by opening the mixer
15. Turn the Low-Mid frequency knob to 20000 Hz
16. Close the EQ and Mixer window

A.9 Consent Form for the Prototype II Controlled Experiment

THE UNIVERSITY OF BRITISH COLUMBIA



November 6, 2002

Human Communication Technologies Lab

Department of Electrical & Computer Engineering
2356 Main Mall, Room 155A

Vancouver, BC, Canada,

V6T 1Z4

Phone: 604-822-4583

Informed Consent Form A

New Interfaces for Expression

Study: Measuring reaction time of three music editing task using

- a) A computer keyboard and mouse combination
- b) A 76 key piano keyboard with a foot pedal
- c) A single point touchpad
- d) A combination of piano keyboard with a foot pedal and a touchpad with the audio turned off
- e) A combination of piano keyboard with a foot pedal and a touchpad with the audio turned on

Project Principal Investigator: Dr. Sidney Fels, Department of Electrical and Computer Engineering, 604-822-5338

Co-Investigator: Mr. Farhan Mohamed, M.A.Sc. candidate, Department of Electrical and Computer Engineering, 604-822-4583. This study is conducted by Mr. Farhan Mohamed as part of his M.A.Sc. thesis in the Department of Electrical and Computer Engineering under the supervision of Dr. Sidney Fels.

Purpose

This study is intended to show how an interactive piano hardware is used in music editing tasks

Study procedure

I will be asked to use the interactive piano prototype to accomplish three music sequencing tasks. The investigators will record my performance and analyze how the system is used, enabling them to determine how it can be improved in future versions. My total participation will be less than 2 hours, which includes training, performing the tasks and filling in questionnaire forms. The investigators insure that the recordings are kept secure in a locked faculty office. All data from me will be coded so that my anonymity will be protected in any publicly available reports, articles and presentations that result from this work.

Confidentiality

I am aware that the responses I make will be recorded. My identity will remain anonymous and my responses will be confidential, known only to the investigators. In any publications that arise from this study I will be identified only by 3 digit random numbers.

If I have any questions about my treatment or rights as a research subject I may contact the Director of Research Services at the University of British Columbia, Dr. Brent Sauder at 604-822-8083.

Consent

I understand that any participation in this study is entirely voluntary and that I may refuse to participate or withdraw from this study at any time.

I have received a copy of this consent form for my own records.

I consent to participate in this study under the above conditions:

Name: _____ Date: _____

Note from the investigators:

We intend for your experience in this study to be pleasant and not stressful in any way. We will be pleased to explain the purpose and methods used in the study to you after your participation has concluded and to furnish you with our results when they are available.

A.10 Consent Form for the Prototype II Case Study

THE UNIVERSITY OF BRITISH COLUMBIA



October 23, 2002

Human Communication Technologies Lab

Department of Electrical & Computer Engineering
2356 Main Mall, Room 155A

Vancouver, BC, Canada,

V6T 1Z4

Phone: 604-822-4583

Informed Consent Form B

New Interfaces for Expression

Study: A case study on using a combination of piano keyboard with a foot pedal and a touchpad for music editing tasks. The following is a brief description of the individual components used.

- a) A 76 key piano keyboard
- b) A momentary foot pedal
- c) A single point touchpad

Project Principal Investigator: Dr. Sidney Fels, Department of Electrical and Computer Engineering, 604-822-5338

Co-Investigator: Mr. Farhan Mohamed, M.A.Sc. candidate, Department of Electrical and Computer Engineering, 604-822-4583. This study is conducted by Mr. Farhan Mohamed as part of his M.A.Sc. thesis in the Department of Electrical and Computer Engineering under the supervision of Dr. Sidney Fels.

Purpose

This case study is intended to show how an interactive piano hardware is used in music editing tasks

Study procedure

I will be asked to use the interactive piano prototype to accomplish music sequencing tasks. The investigators will perform video recordings of my performance and analyze how the system is used, enabling them to determine how it can be improved in future versions. My total participation in the case study will be 2 hours per day for a total of 7 days. The investigators insure that the video recordings are kept secure in a locked faculty office. All data from me will be coded so that my anonymity will be protected in any publicly available reports, articles and presentations that result from this work.

Confidentiality

I am aware that the responses I make will be recorded. My identity will remain anonymous and my responses will be confidential, known only to the investigators. In any publications that arise from this study I will be identified only by 3 digit random numbers.

If I have any questions about my treatment or rights as a research subject I may contact the Director of Research Services at the University of British Columbia, Dr. Brent Sauder at 604-822-8083.

Consent

I understand that any participation in this study is entirely voluntary and that I may refuse to participate or withdraw from the case study at any time.

I have received a copy of this consent form for my own records.

I consent to participate in this study under the above conditions:

Name: _____ Date: _____

Note from the investigators:

We intend for your experience in this study to be pleasant and not stressful in any way. We will be pleased to explain the purpose and methods used in the study to you after your participation has concluded and to furnish you with our results when they are available.

A.11 Comments on the Questionnaires

THE UNIVERSITY OF BRITISH COLUMBIA



Human Communication Technologies Lab

Department of Electrical & Computer Engineering
2356 Main Mall, Room 155A
Vancouver, BC, Canada,
V6T 1Z4
Phone: 604-822-4583

KEYed User Interface Prototype I Comments

Some of the comments on the questionnaire which was provided during the experiments with KEYed user interface prototype I are as follows:

1. *A good system as it saves time, eases the efforts of musicians and puts two input devices into one.*
2. *Within a year there will be two in every American Home.*
3. *Very exciting new way to control: nice linkage between music playing and editing.*
4. *Easy to use.*

KEYed User Interface Prototype II Comments

Some of the comments on the questionnaires which were provided during the experiments with KEYed user interface prototype II are as follows:

1. *I like the musically arranged editing*
2. *Touchpad is difficult to use, although better while holding the piano keys when using it.*
3. *Suggestion: Use tunes or chords for general functions to make editing more enjoyable.*
4. *I like it because I don't have to reach for the mouse and keyboard.*
5. *Didn't like hearing notes in Edit mode.*
6. *I find touchpad too small.*
7. *For using the touchpad, I prefer the notes to access the touchpad functions to be on lower octaves.*

8. Suggestion: Adding a visual indicator for the octaves, not necessarily individual notes. Either color or a note layout system on the piano keyboard.

9. Once I practice the key mapping, the whole task becomes very simple indeed.

10. I find the current note sounds while editing distracting. Although I prefer having sounds, maybe a different sound?

11. I like not having to switch input devices and hence find it faster.

12. Perhaps different sounds could be triggered during editing.

13. I find touchpad hard to use as a single device.

14. Very exciting idea, a natural solution for pianist composers.

15. Easy to use.

16. The foot pedal is easy to use as I am used to it.