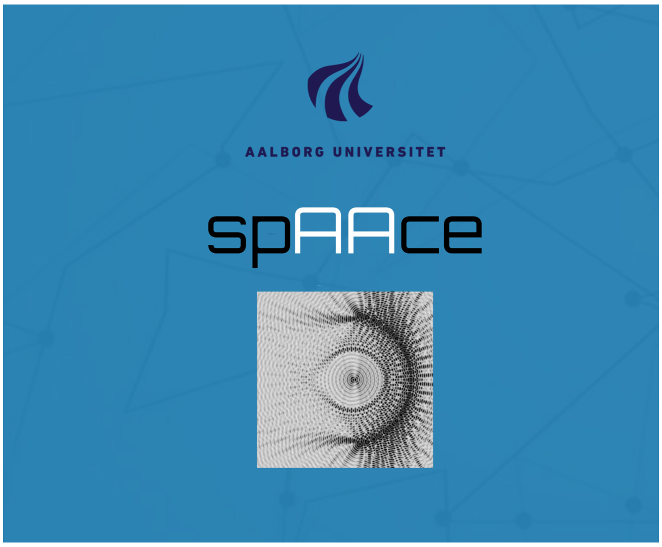

spAAce: a system for spatial music improvisation

Project Report

Jorge Madrid Portillo, Diego Di Carlo, Matteo Girardi

Aalborg University Copenhagen
Sound and Music Computing

Sound and Music Computing
Aalborg University Copenhagen
http://www.aau.dk

AALBORG UNIVERSITY
STUDENT REPORT

Title:
spAAce

Theme:
Sonic Interaction Design

Project Period:
Spring Semester 2016

Project Group:
group n.5

Participant(s):
Jorge Madrid Portillo
Diego Di Carlo
Matteo Girardi

Supervisor(s):
Dan Overholt
Francesco Grani

Copies: 1

Page Numbers: 67

Date of Completion:
May 27, 2016

Abstract:

We present a system called *spAAce* which let users control real-time movements of sound sources by drawing trajectories and manipulating them with hand gesture. The first prototype of this application has been developed for Wavefield Synthesis Systems together with WFSCollider, an open-source software based on Supercollider which let users handle such synthesis technique. In order to communicate with such software the Open Sound Control protocol has been used. The *spAAce* application has been implemented using Processing, a programming language for sketches and prototypes within the context of visual arts. This application aims to let sound artists improvise with spatial music in live performance and live electronics.

AALBORG UNIVERSITET
STUDENTERRAPPORT

Lyd og Musik
Aalborg Universitet København
http://www.aau.dk

Titel:
spAAce

Tema:
Sonic Interaction Design

Projektperiode:
Spring Semester 2016

Projektgruppe:
group n.5

Deltager(e):
Jorge Madrid Portillo
Diego Di Carlo
Matteo Girardi

Vejleder(e):
Dan Overholt
Francesco Grani

Oplagstal: 1

Sidetal: 67

Afleveringsdato:
27. maj 2016

Abstract:

We present a system called *spAAce* which let users control real-time movements of sound sources by drawing trajectories and manipulating them with hand gesture. The first prototype of this application has been developed for Wavefield Synthesis Systems together with WFSCollider, an open-source software based on Supercollider which let users handle such synthesis technique. In order to communicate with such software the Open Sound Control protocol has been used. The *spAAce* application has been implemented using Processing, a programming language for sketches and prototypes within the context of visual arts. This application aims to let sound artists improvise with spatial music in live performance and live electronics.

Contents

Preface

This report has been written in May 2016 as part of the semester project at Aalborg University of Copenhagen under the supervision of Dan Overholt and Francesco Grani. We thank a lot our supervisors for valuable assistance and wise guidance. During the project, we developed a Processing–software for musical performance of spatial music. The software is meant as a controller and let the user draw rails and paths which are used to move sound sources in a virtual auditory environment. Using a Leap Motion interface the program can recognize some particular gestures which let the user interact with the sound sources. In order to render such virtual environment, the program makes use of Wave Field Synthesis technique and WFSCollider software as sound rendering engine. Even if the software we developed is merely a prototype, our project aims to enhance the overall performance of spatial music. Such goal is far to be achieved since many improvements and refinements can be done, however, this project will hopefully establish interesting foundations for future improvement of spatial music improvisation in live electronics scenarios.

Aalborg University, May 27, 2016

Jorge Madrid Portillo
<jmadri15@student.aau.dk>

Diego Di Carlo
<ddicar16@student.aau.dk>

Matteo Girardi
<mgirar15@student.aau.dk>

Chapter 1

Introduction

During the second part of the 20th century, composing music with new technologies has constantly become more relevant, and *electro-acoustic music* has unveiled new future developments. The timbre, as musical parameter, has been widely studied so as to explore new musical spheres. Another important musical parameter has gained more and more interest among avang-gard composers, sound artists, sound engineers, movie industries and universities: it is the *musical space*. Indeed, the spatial characteristics of a composition have been an important topic in the context of avant-gard music in the past decades [40] and it is still a relevant quality of a musical piece [2]. From an artistic point of view, conveying spatial musical ideas and thoughts could underestimate the technical issues that must be faced during the development and implementation process of a software for musical purposes. Hence, contemporary composers and sound engineers have to find a trade-off during such process of composing new musical material. Moreover, learning new technologies or software for spatial music could be time consuming for composers that do not have a deep knowledge in the computer music field. Therefore, we have developed a graphic application called *spAAce* which attempts to provide the following advantages:

1. A quick way to sketch and test movements of sound sources.

2. Improvisation of spatial music during live performances.

The latter point could contribute to enrich performance of music concerts in a live electronics scenario. Taking advantage of a peculiar spatial technique as Wave Field Synthesis and using WFSCollider as sound engine rendering software, the *spAAce* application lets composers focus on creating trajectories for sound sources and interacting with them in real-time. For this reason we have focused our development on the design of a graphical user interface that helps composers to spatialize sound in a more expressive and direct way.

1.1 Context

This application has been developed as semester project of Sound and Music Computing Master's programme at Aalborg University Copenhagen. The main theme of the semester was Sonic Interaction Design and has been supervised by Dan Overholt and Francesco Grani.

1.2 Motivation and Goals

The fundamental motivation for developing such application is to give composers a tool that enables them to perform real-time spatialization so as to improvise with their musical materials during live performance. Indeed, there are several spatialization applications, however we believe that they are not easily manageable to improve spatial music performances. Hence, our intention is to research if is there any possibility to let composers express their musical intentions. The idea was firstly motivated by Matteo Girardi, who made several electro-acoustic music experiences with Alvise Vidolin and found that a spatialization tool could be developed in order to satisfy the composers necessities for spatial music improvisation. The application is far to be released as stable software, therefore, *spAAce* is to be considered as a prototype in order to test our hypothesis.

Goals

- Platform independence: Ideally any 3D audio rendering algorithm on any computer platform should technically be able to understand *spAAce*;
- Easily understandable syntax: to prevent misunderstandings when stored data are shared;
- Extendability: Easy adding of descriptors to extend the specification especially as long as *spAAce* is in development;
- Free and open source: to increase the acceptance and widespread usage of this new format;
- Easy to connect: with interfaces, controllers and sensors for real-time control of spatialization;
- Use of existing standards: to focus on conceptual rather than technical development.

Thesis

To design and develop a multimodal interactive and real-time system that lets the composer, artist and sound engineer perform spatial music and makes it more transparent for the audience.

Hypothesis

- We assume WFS technique and the rendering engine work correctly, which means the representation of sound sources in space is also correct.
- Our goal is not to measure the absolute sonic perception of the audience, but only the user performance comprehension.
- The mixing is separated from the spatialization, so that there is a one to one relation to each of the sound sources.
- sound effects are not considered.
- The users/audience is in the center of the rendering system.

1.3 Project overview

The main concept of *spAAce* is to design and develop an application which provides composers with several multimodal interaction tools for real-time music spatialization. The system is designed to be used during live concerts that use modern surround systems such as *WFS* or *Ambisonics*. Such multimodal interaction tools are divided in :

- pen gesture interaction
- hand gesture interaction

The implementation has been done in order to move sound sources within a predefined space (inside and outside the loudspeakers array) and to draw several sound source paths (line, circle and free shapes). The pen interaction is the primary input of the software and all the details will be explained later in the report. The hand gesture interaction has been implemented using a Leap Motion interface. The design of its interaction will be explained in further sections. The target users of the application are composers, sound artists and sound engineers having previous knowledge about 3d sound systems preferably, but other users can be easily trained to perform a spatial composition. The application was implemented in Processing.


spAAce
Real Time Spatialization
KØBENHAVN
AALBORG UNIVERSITET

Chapter 2

State of the Art

Within the context of audio spatialisation, several spatial techniques have been implemented such as VBAP [33], DBAP [24], Ambisonics [26] and Wave Field synthesis (WFS) [39]. These spatial audio rendering techniques are the current state of the art regarding sound spatialisation. Moreover, several software has been developed recently such as Spat [22], WFSCollider [7], Wonder [3] as well as several libraries [14] [36] [38]. Using such software for audio spatialisation, composers, sound artists and sound engineers are able to move sound in a virtual sonic environment, to convey their musical intentions within the spatial domain and to enhance their musical ideas. We believe that they are indeed powerful tools, however, they lack of a proper interface that a composer or an artist can intuitively use to convey their creativity. With the spreading of user-friendly GUI development environments for mobile and web app, lots of client application have been developed for these rendering engines, which allow real-time finger-based interaction. Some of them are more mixing-oriented, providing a real-time positioning of the sound sources in a virtual environment, while others, such as Trajectoires [10] and Spatium [31], allow users to move sound sources and create complex paths both in time and in space. However, these new applications are still in the embryonic stage and there is still lots of work to do in order to design the most suitable interface that can capture the actual intentions of performers.

2.1 Human Computer Interaction

Early and definite examples of the synergy between music and HCI can be found in the research and contributions undertaken by William Buxton during the 1970s and 1980s; it was in fact, the design and use of computer-based tools for music composition and performance, which led Buxton into the area of HCI. Buxton has even ironized that there are three levels of design: standard spec., military spec., and musical or artist spec, being the third the hardest and most important. From the musical side, several computer music researchers have studied the control of sound in musical instruments as well as aspects of the communication between players and their instruments. Pressing studies and compares the sound control issues of a violin and a standard MIDI keyboard using no less than ten dimensions. Vertegaal and Eaglestone evaluate timbre navigation using different input devices. Wanderley proposes a basic gesture taxonomy evaluating how such gestures relate to different sound control tasks, and approaches the evaluation of input devices for musical expression by drawing parallels to existing research in the field of HCI [21].

NIME - New Interfaces for Musical Expression

The history of NIME (New Interfaces for Musical Expression) started out in 2001 is Seattle, Washington as a worshop at the ACM Conference on Human Factors in Computing Systems (CHI). Since then, an annual series of international conferences have been held around the world, hosted by research groups dedicated to interface design, human-computer interaction, and computer music. The International Conference on New Interfaces for Musical Expression gathers researchers and musicians from all over the world to share their knowledge and late-breaking work on new musical interface design [1].

Introduction of NIME theory

Advances in digital audio technologies have led to a situation where computers play a role in most music production and performance. Digital technologies offer unprecedented opportunities for the creation and manipulation of sound, however the flexibility of these new technologies implies a confusing array of choices for musical composers and performers. Some artists have faced this challenge by using computers directly to create music and leading to an explosion of new musical forms. However, most would agree that the computer is not a musical instrument, in the same sense as traditional instruments, and it is natural to ask 'how to play the computer' using interface technology appropriate for human brains and bodies [12].

NIME evaluation

Evaluating a NIME is not an easy task since there could be many perspectives from which to view the effectiveness of the instruments we build [29]. Performing on an instrument is indeed a means of evaluating how well it functions. Moreover, the audience response to their performance is a measure of success and audience evaluation is based on how they feel during a performance; in other word how engaged they are. It is common to describe a good performance as "exciting",

[1] `http://www.nime.org/`

"skillfull", "musical" otherwise it would be described as "boring". Hence, for the field of DMI design the term "evaluation" deserves a much broader definition than that typically used in human-computer interaction (HCI). If we consider that we can extended the stakeholders to composers, instrument builders, component manufacturers and even customers, which may have a different concept of what is meant by "evaluation", then the task of evaluating a DMI is getting much more complicated.

2.2 Artistic Spatialization Practice

Introduction to sound and music in space

The practice of positioning and spreading sound sources in an environment such as a room, a concert hall, an auditorium or a church is not new in the western music scenario. If we look throughout the history of western music we can find many examples of such practice. For instance, Charles E. Ives, considered as the leading American composers of art music of the 20th century [35], paid attention toward music in space, its perception and its distribution. As he wrote in his notes of the *4th Symphony*:

> "Experiments, even on a limited scale, as when a conductor separates a chorus from the orchestra or places a choir on the stage or in a remote part of the hall, seem to indicate that there are possibilities in this matter that may benefit the presentation of music, not only from the standpoint of clarifying the harmonic, rhytmic, thematic material, etc., but of bringing the inner content to a deeper realization. [20]"

There are others examples of such thinking in his compositions, as in *Central Park in the Dark* and *The Unanswered Question*. In the latter, the musicians are spread around and such practice has not been ignored by composers of the 20th century. Indeed, Henry Brant, another American composers, has followed such habit in composition as *Antiphony I* and *Voyage Four*. However, looking even back into the history of western music, we can find many others examples where the physical position of musicans played an important role. Composers as Adrian Willaert, Monteverdi, Mozart, Berlioz, Wagner, Mahler, Schönberg and Satie have specified musicians position on the stage. Looking at more recent history, particularly to electro-acoustic music, one can find an outbreak of spatial music due to new technologies as micro-processor, computers, loudspeakers and audio interfaces which let composers, sound engineers and sound artists extend their musical ideas.

Karlheinz Stockhausen is probably one of the first composers who utilizes such new means to compose musical pieces. In his *Gesang der Jünglinge* (1955/56) new musical horizons are unveiled:

> "The old indivisible relationship between music and space here receives a new impetus, movement of the sound, for the sounds continuously 'wander' from one to another of the loudspeakers placed around the audience. [42, p. 160]"

Direction and movements of sound within space had been determined by the composer as a new dimension of the musical experience [42, p. 41].

Luigi Nono is another composer whose compositions have highly taken advantages by new technologies for musical purposes. He is probably the composer who had carefully deepened the space as new musical parameter. The electro-acoustic technologies have surely increased the possibilities of using this feature and it is indeed an important field of development.

2.3 Overview of Spatialization Techniques

Surround 5.1/7.1

Surround is an audio technology that use more than two loudspeakers so as to surround the listener. In 1982 the Dolby laboratories creates the well known 'Dolby Surround' which has been massively used in cinemas and 'home theater' systems. There are several standard configuration of such system, for instance, surround 5.1, 7.1 and 10.1. The former is to be considered as an enhancement of stereo panning, indeed, the audio signal is elaborated by a matrix which calculates the amplitude of each channels based on the loudspeakers position [18].

Binaural audio

Typically, binaural audio allows the finest and most accurate audio reproduction since the audio signal can be digitally processed so as to match carefully the natural sound perception of a given person.

> In its purest form, binaural reproduction aims to reproduce all of the cues that are needed for accurate spatial perception, but in practice this is something of a tall order and various problems arise [34].

Binaural audio can only be listened to through earphones. Moreover, to have a correct binaural audio reproduction the 'Head Related Transfer Function' must be applied, that is catching the spectral differences of frequencies and time between the two ears of the listener. To record binaural audio it is often used the so-called 'dummy head' where microphones are placed inside gummy ears.

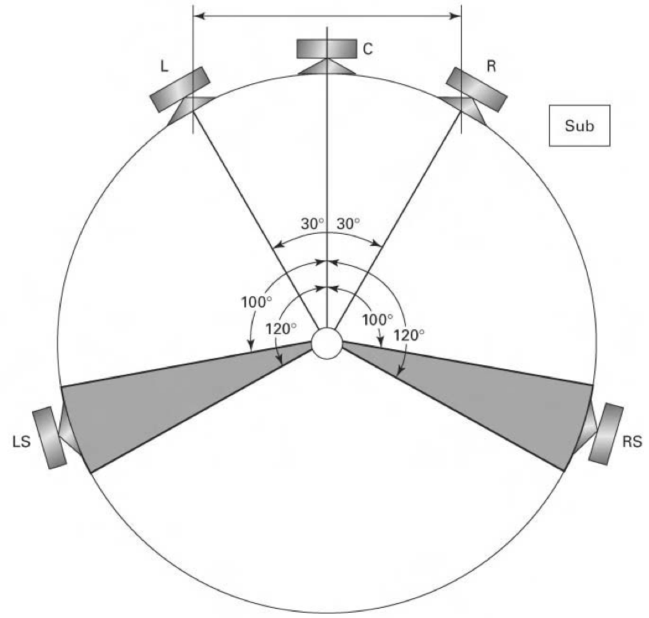

Figure 2.1: Surround 5.1

Figure 2.2: Binaural microphone Neumann KU100

Vector Base Amplitude Panning (VBAP)

This spatial audio rendering technique has been implemented by Ville Pukki, and it is an extension of the panning technique to an array of loudspeakers. Theoretically, it allows an infinite numbers of loudspeakers in a 3-dimensional or 2-dimensional configuration so as to surround the listener [33]. In this technique the 'amplitude panning' method is reformulated with vectors and vectorial base. Compared to

Surround technique, VBAP provides several advantages such as:

- the number of loudspeakers can be decided in advance
- the position of loudspeakers can be decided in advance
- the acoustic interference is minimized

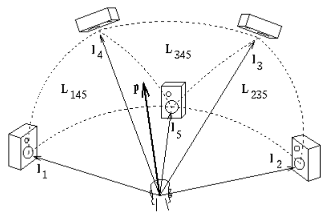

Figure 2.3: VBAP Loudspeakers configuration

Distance-Based Amplitude Panning (DBAP)

Typically spatial audio rendering techniques define a 'sweet spot' inside the loudspeakers array where the listener should be located so as to have a correct sound perception. However, in live performance and concerts scenarios, such layout goes against the logic of conveying musical contents to a wide number of listeners. Distance-based amplitude panning (DBAP) offers an alternative panning based spatialization method which does not assume a specific location of the listener nor a specific loudspeakers position. Distance-based amplitude panning (DBAP) extends the principle of equal intensity panning from a pair of speakers to a loudspeaker array of any size, with no prior assumptions about their positions in space or relative to each other [24].

Ambisonics

Ambisonics is a powerful technique for sound spatialization. It can allow recording, manipulation, and composition with naturally and artificially constructed three-dimensional sound fields [26]. The Ambisonic surround-sound system is essentially a two-part technological solution to the problems of encoding sound directions (and amplitudes) and reproducing them over practical loudspeaker system so that listeners can perceive sounds located in three-dimensional space. This can occur over a 360-degree, horizontal-only sound stage (pantophonic systems), or over the full sphere (periphonic systems). The system encodes signals in B-format which contains three channels for panto-phonic systems and a further channel for periphonic, i.e., with-height reproduction. These signals convey directionally encoded information with a resolution equal to first-order microphones (cardioid, figure-eight, etc.). Reproduction requires four or more loudspeakers depending upon the required reproduction (pantophonic or periphonic) and on the size of the performance area. Practical minimal are four if the sounds are limited to the horizontal plane, and eight if height is required. It is important to note that it is not necessary to consider the actual details of the reproduction system during the original recording or synthesis of a sound field. The only exception to this is that the vertical dimension is essential if a width-height re-play system is required. If the B-format specifications are followed, assuming suitable loudspeaker/ decoder systems are used, then operation in different venues will be as similar as local acoustics allow. In all other respects the two parts of the system, encoding and decoding, are completely separate. Ambisonics provides significant advantages in that the effect does not sound significantly listener- or speaker-dependent (you can even walk outside the speakers and appreciate the image). In addition, listeners can position the loudspeakers (within fairly wide limits) in any convenient position in the room.

Wave Field Synthesis

Wave field synthesis (WFS) is a spatial sound field reproduction technique that utilizes a high number of loudspeakers to create a virtual auditory scene over a large listening area [39]. Contrary to traditional spatialization techniques such as stereo or surround sound, the localization of virtual sources in WFS does not depend on, or change with the listener's position. There is not a specific 'sweet spot' where the listeners should be positioned, but a larger area called 'sweet zone'. The theoretical framework of WFS was formulated by Berkhout et al. at Delft University of Technology almost 20 years ago [6].

WFS let a large amount of listeners have a correct spatial perception. In order to fully take advantage of this technique, the place should be acoustically treated so as to avoid reverberation and reflections. There are some disadvantages too, such as high number of loudspeakers, several computer and audio interfaces to run the whole system, which means high costs. There is a wide range of applications where this technique can be used, such as cinemas, theaters, sound installations and concerts. Currently there are few WFS systems available, and they can mainly be found in academic environment.

2.4 Overview of Spatial Sound Rendering Engines

Since the aim of this report is not to describe all the sound rendering engines available, we will focus on the spatial sound rendering engine we used throughout the development of *spAAce*.

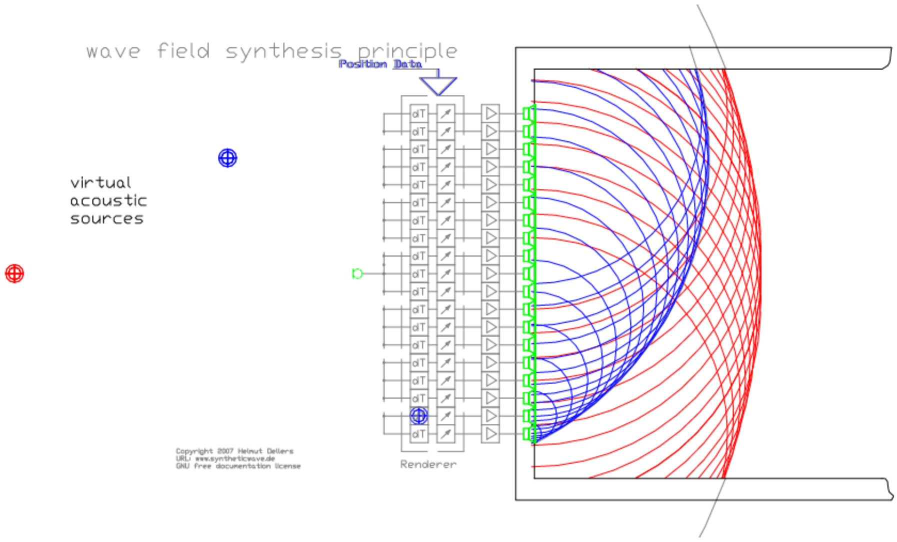

Figure 2.4: Wave Field Synthesis

WFSCollider

Currently, there are quite few Wave Field Synthesis softwares and unfortunately most of them are not free. WFSCollider stand out from them, it is free and open source. WFSCollider has been developed by Game of Life, a dutch foundation, and it is based on SuperCollider. This software is a powerful tool for managing this technique and provides many different features. WFSCollider can create sound sources using synthesis technique provided by the software or by importing sound files, then several trajectories are available so as to move them in a two-dimensional space. Due to its intuitive GUI, it is easily understandable, hence, it is not necessary to know how to program with SuperCollider. However, one can still use it to control WFSCollider engine. It is possible to create compositions 'offline', without having a WFS system ready to use. This software can export composition project for many loudspeakers setup as stereo, quadriphonics, octophonics and even binaural audio for earphones.

2.5 Application for spatial sound scenes descriptors

Review of similar spatialization software interface

In the following sections we present similar spatialization software.

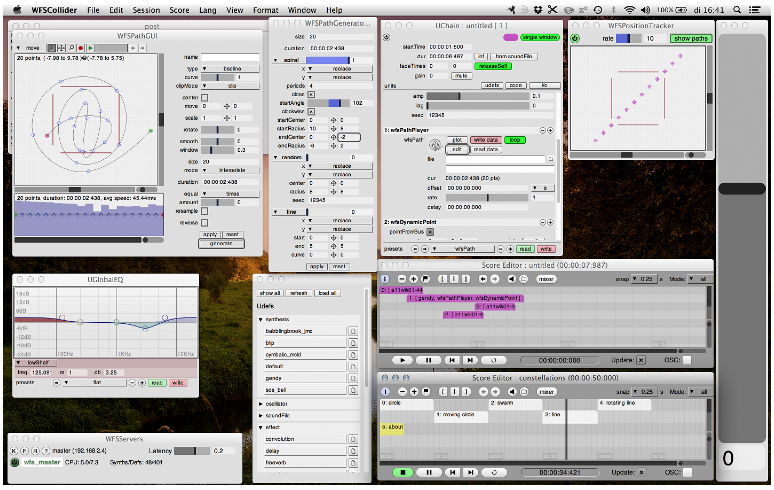

Figure 2.5: Screenshot of WFSCollider, the sound rendering engine for Wave Field Synthesis spatialization based on SuperCollider Image courtesy of `https://sourceforge.net/projects/wfscollider` (27/05/16)

Spatium

Spatium [32] is a software for sound spatialization written by Rui Penha and Joao Pedro Oliveira. It consists of a set of free, open source and modular tools providing spatialization renderers, spatialization interfaces, DAW plugins and Max objects that communicate via OSC. As the authors say, the aim of *spatium* is to facilitate the exploration of different approaches to sound spatialization, ease the integration of sound spatialization into diverse compositional work flows, smooth the transition from the studio to different performance environments and be easily expandable to cater for growing needs. The authors propose several advantages by using *spatium*, such as:

- simple integration with DAWs;
- easy adjustement to different studio and performance environments;
- ability to choose the most suitable interface paradigm for each musical intention;
- real-time use of sound spatialization through intuitive interfaces;

Such modular approach is a fruitful solution so as to integrate sound spatialization into different compositional needs and to allow different combinations of rendering algorithms and controlling interfaces. As an open source project, it may also benefit from a community development and, given the modular structure, one can even integrate the proposed interfaces with their own spatialization renderers.

spaceJam

spaceJam has been developed as an option to authoring software since many spatial sound artists are currently experiencing major creative setbacks [25]. This software is an interactive multimedia authoring tools designed for the 3D audio spatialization. The GUI architecture is a non-modular, multi-frame view that attempts to guide users effortlessly through the sound spatialization procedure. Users are presented with three perspectives of a 3D virtual model that intentionally mimics their real room and up to 256 speaker channels, of which they may pan up to 256 virtual sound sources within. Gesture capture of sound source spatialization is a key feature of this tool, along with multi-channel audio file recordings that are based on source spatialization.

Trajectoires

Trajectoires is a web application running on desktop computers, smart-phones or tablets that lets composers draw trajectories of sound sources to remotely control any spatial audio renderer using the Open Sound Control protocol. Interviews and collaborations with contemporary music composers helped to inform the design of the tool, assess it in real music production context and add several features to manipulate existing trajectories [11]. *Trajectoires* is implemented in JavaScript and HTML running on the many devices with a web browser. This application provides several features such as: drawing, playing and editing sound source trajectories, communication with Spat engine [22], having simultaneously several simulation clients and reducing the number of trajectory points. Users can define the orientation of the sound source at specific positions of an existing trajectory using physical manipulation of the mobile device. *Trajectoires* allows to store users' work in sessions, to navigate thought them and to export the data.

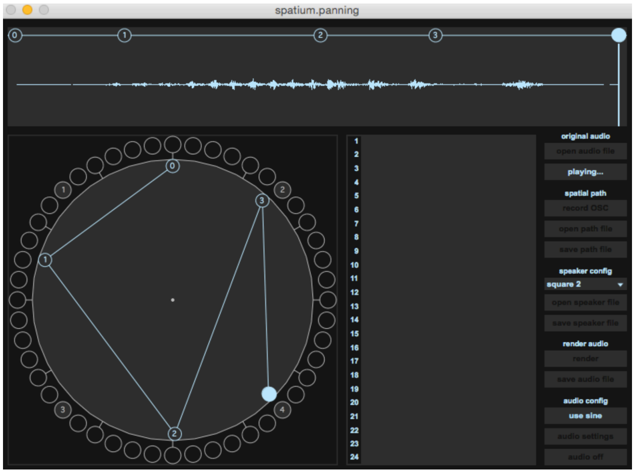

Figure 2.6: Screenshot of *spatium*, a *Cycling'74 Max* tool for sound spatialization. Image courtesy of `http://spatium.ruipenha.pt/` (27/05/16).

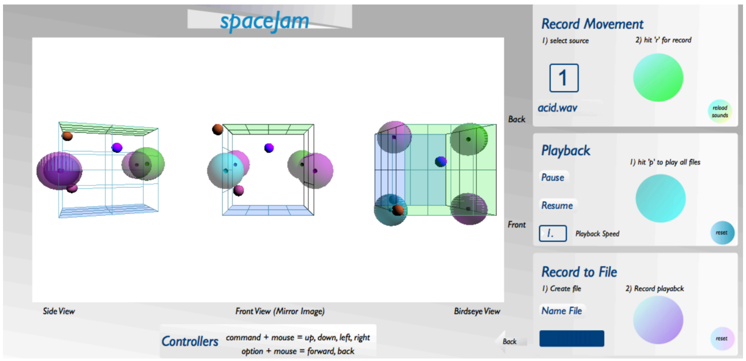

Figure 2.7: Screenshot of *spaceJam* [25], a software for 3D sound spatialization. Image courtesy of `http://adrianamadden.com/spacejam/` (27/05/16.)

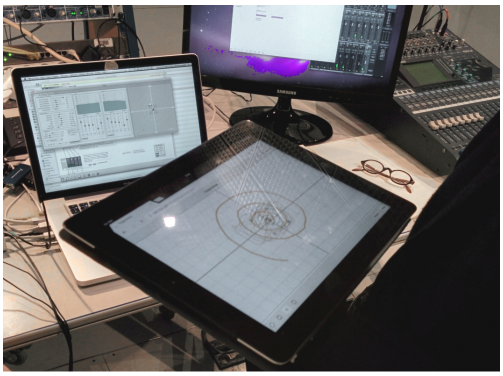

Figure 2.8: An artist using *trajectroies* [11], the graphical interface for *SPAT*, the sound rendering engine developed by IRCAM. Image courtesy of [11].

Chapter 3

spAAce

3.1 Overview of the System

The system is designed to provide full control over the spatialization of multiple sound sources in a real-time context. For this matter, there are multiple constraints that force the users to use particular hardware and software:

- **Rendering system**: The current technology is mostly divided between either WFS or Ambisonics since they seem to be the two best options when trying to perform audio spatialization. The data collected from the interviews suggest that WFS system is probably being used more in terms of audio spatialization for composers and music performances, this is mainly the reason why spAAce is developed to work with WFS as hardware and WFSCollider as rendering system, but would be theoretically perfectly possible to develop an extension in order to make it work with any other system available.

- **Tactile surface controller (WACOM CINTIQ 22")**: Since spAAce has been developed for processing, it is possible to export the project into other environments rather than PC platforms like Android, but this is of course not trivial. Having this is mind the best solution is to run the application in a PC, which is then connected to a tactile display like WACOM surface, which is the one that was chosen for the tests and demo performances.

- **Gesture interaction**: There are few options in the current time when it comes to hand gesture detection, LEAP Motion is leading the industry for the simplicity of use and the available API's, for this reason we decided work with it in order to implement the gestural controls and gesture interaction with the system.

Apart from these constraints the system is assumed to be under the following hypothesis:

- The performer has knowledge about rendering system and spatial audio
- There is in fact a rendering system to connect to

3.2 History and Progress

The project started as an idea by Matteo Girardi to create a system which would enable new avant-garde composers to create real-time audio spatialization improvisations. Soon, after reading the literature on spatial art and performance, we fixed our ideas under the constraints and hypothesis described. The design process started by conducting a series of interviews with the composers, specially with Alivise Vidolin. These gave us a lot of insight on how to start developing the application. The first prototype was developed under one month of development and tested in the AAU multi-sensory lab. The first test reported good feedback on interaction but also plenty bugs and features which would improve the performance and expressiveness of the app. The second prototype was finished after other two months of development, around May 2016. Then final the final tests, audiovisual material and report were done in the last days of May 2016.

Project Setup

The project was developed under the supervision of both *Francesco Grani* and *Daniel Overholt*. We used Trello.com®for task management and Google Drive®for file storage. The code was versioned with git using a Github repository.

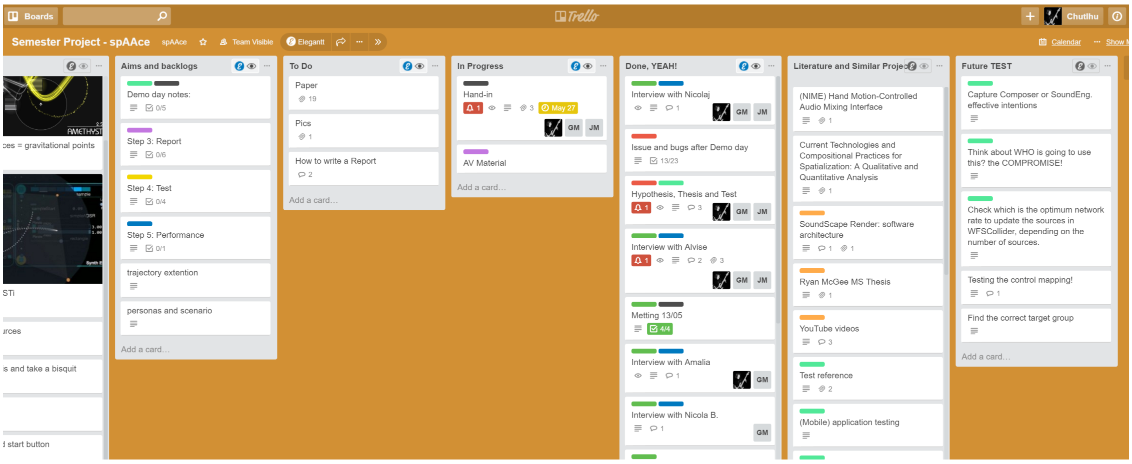

Figure 3.1: A screenshot of our *board* on *Trello.com*®, a web-based project management application (27/06/16)

3.3 Philosophy and metaphors

In order to develop our application we applied the *railway metaphor*. The railway metaphor means that the user can create trajectories and then apply sound sources to them, same way as train railways, so that multiple sound sources can follow the same railway in different directions at different speeds.

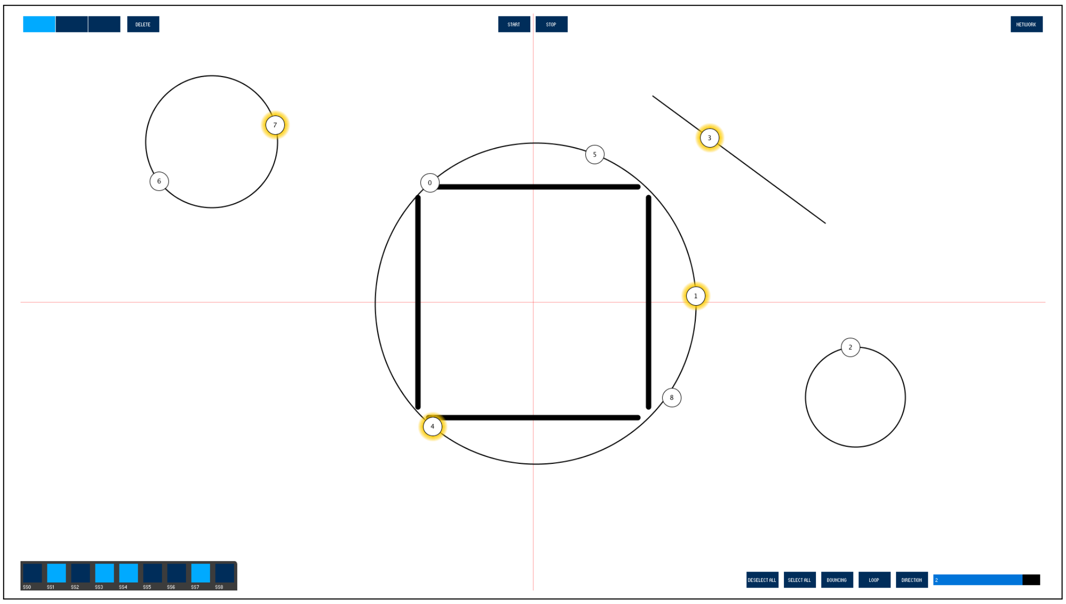

Figure 3.2: *spAAce* processing application example

3.4 Terminology

Spatialization The art or technique in which the artists or technician uses the software available to put the sound sources in the space to create a particular effect or atmosphere for the audience.

Sound Source A source in space which emits a particular sound. This sources are linked to the mixer so that each of the sources represent a particular track.

Trajectory The path that a certain sound has to follow.

Gesture A motion or movement the user has to perform in order to interact with the application.

3.5 Physical setup

All the experiments have been conducted with the following set up (see Figure 3.3):

- **WFS**: a WFS system of a square of 16 loudspeakers for each side of the room.
- **Computer**: a computer running WFSCollider server for sound rendering (the OSC server).
- **Laptop**: a laptop running the Processing sketch spAAce, placed near the center of the WFS system and connected to the server via Ethernet (the OSC client)
- **Touchscreen**: a Wacom Cintiq 22″ touch standing in the center of the system as a input/output device for the client.

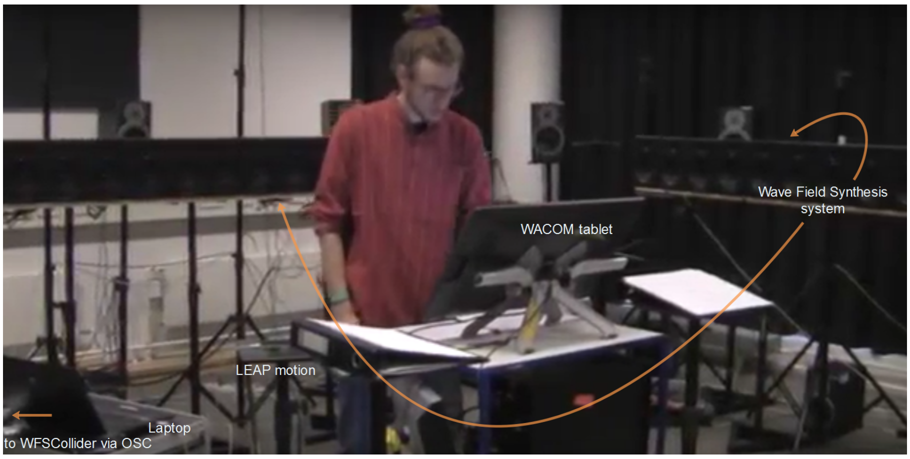

Figure 3.3: Physical setup in the multi-sensory lab of Aalborg University in Copenhagen.

Chapter 4

Design

4.1 Composers and Sound Engineers interviews

Before starting to design and implement this project, having direct feedback from composers was a suitable starting point. Therefore, we did some interviews asking what are their thoughts regarding spatial music, what are their needs, what kind of interaction they would prefer to have, what features and tools they would need while improvising spatial music. This was a required step so as to gather proper information.

The interviews were conducted at Aalborg University Copenhagen via Skype or in a face to face meeting. The aim of the interviews was to obtain as much information as possible before starting the implementation process in order to satisfy the user requirements. These precious information allowed us to develop better interaction and to decide which trajectories should be used. In this section we will discuss briefly some of the relevant information extracted from the interview translations.

Interview with Nicolaj Kynde

According to Kynde there are two possible ways of using such system. Firstly, the performer can create a real-time spatial mix. Secondly, he can create an abstract performance to explore more artistic or aesthetic approaches. We also extracted some knowledge in terms of shape design and trajectories, for example some concept as shape morphing and mixing trajectories with each other.

Interview with Nicola Bernardini

From Bernardini's interview we have extracted the idea that visual feedback con help composers to create their compositions and take them from the theoretical to the practical world. In that case, the composer can communicate to the performer how the spatialization should be like, who will then perform it using *spAAce*. Again Bernardini said something related to what Nicolaj said in his interview, which is the possibility to use *spAAce* as a sketching application to create compositional prototypes or to explore and test different spatial solutions.

Interview with Amalia De Götzen

In her interview De Götzen described spatial composition as a way to clarify the musical texture. According to De Götzen, it is very difficult to determine a symbolic patterns, each composers have his own way to portray spatial movements of sound and a standard cannot be found. She found the idea of spatial music improvisation really interesting and she provided us with some examples. However, such examples were specifically created for that particular performance. Moreover, she believes that our application will not be used by composers since usually they do not perform, hence, it would be more useful for sound artists.

Interview with Alvise Vidolin

According to Vidolin, one required feature is to have several basic shapes like lines, circles an free hand drawing. Moreover, if such shapes are intended to be path and rails it is therefore necessary to define velocity and acceleration for each sound source. Once there are many sound sources, it is useful to have a global control of these parameters. Vidolin gave us many valuable hints on how to design the interaction with these parameters. As stated by Vidolin there are two plans of action, such as:

- gestural interaction
- physical interaction

The first could be some kind of sensors attached to the performer hands or a Leap Motion, for instance. The latter could be a midi controller. The most easy way to draw trajectories and to visualise a two-dimensional space is using a tablet surface. Vidolin proposed to perform a milestone piece of electro-acoustic music as *Turenas* by John Chowning, for instance.

4.2 Design Framework

NIMEs often fall into the category of electronic and digital musical instruments (DMIs)[37]. However, since the interface and sound unit are separable, evaluation can be done on different level of granulate by evaluating distinct elements of a whole system with different goals [37].

We believe that our system could be considered a New Interface for Music Expression (NIME), since the purpose of this project is to design and develop a new interface for moving sounds in place, enhance the composer or the sound-engineer performance and compose spatial music.

In order to account for those demands, a handful of evaluation framework have been introduced [29], [28]. In particular we found the MINUET design framework [28] a useful and complete framework that cover both the designing and the evaluation processing. It guided our designing by helping us in deciding which evaluation criteria could be important with reference to the possible perspective of interest.

This model combines the studies in the field of Human-Computer Interaction (HCI) with focus on User Experience (UX), both of the performer and the user. As shown in figure 4.1 MINUET is structured as a design process consisting on two stages: the first stage, *Goal* describes the objectives of the interface, while the second stage, **Specifications** helps designing the interaction in order to fulfill there objective [28].

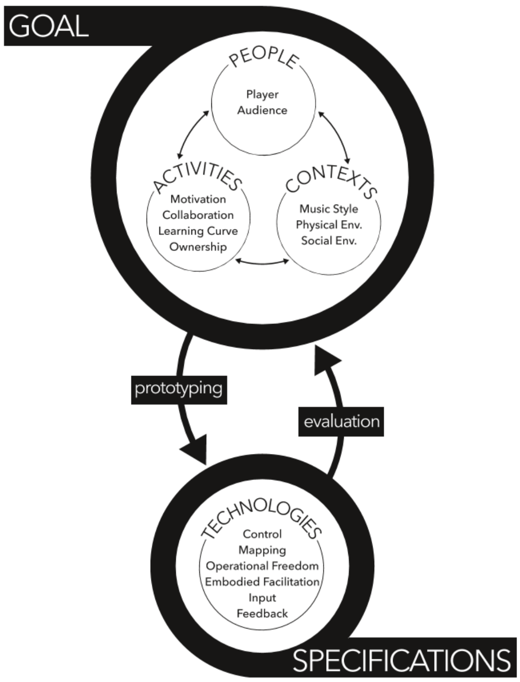

Figure 4.1: MINUET framework. Image courtesy of [28]

The design process starts from the analysis of the designer *goal*, which describes the purposes of interaction. This stage takes into account three aspects: **People**, which are end-user (*who*), **Activities**, specifying the kind of interactions the designer has in mind (*what*) and **Contexts** detailing the specifications of the environment (*where* and *when*) [28]. Once the interaction goals have been defined, designers can move on to the *Specifications* stage by prototyping the interface. This stage analyses how to design the interaction resembling the last entity of framework, **Technologies** [28]. Lastly, designers can evaluate the proposed specifications by referring back to the original goals.

This process suite very well with the well-known framework called *Agile project management*, which is an iterative, incremental method of managing the design and build activities of engineering, information technology and other business areas that aim to provide new product [16].

According to this strategy, we chose the MINUET framework as the base line of an iterative process, so that we reached the final prototype simply iterating this work organization.

A MINUET for *spAAce*

Here we briefly summarize our system according to the MINUET perspective:

Goal

spAAce is a multi-modal and real-time interactive system that lets the user improvise with the spatial dimensions of the sounds in a way that can be more comprehensible by the audience.

People

Artists, Composers that want to add the component of the physical space at their composition or performance. The role of the *audience* is quite relevant, because this system should add more transparency to the performance.

Activities

spAAce is designed to let the performer to assign each sound (sources) to the illusion of location in a specific place in the space. *Motivation:* With the spreading of virtual acoustic environment and multi-channel sound system for concert halls and auditoriums and due the lack of a proper interface to perform spatial composition, such a system could be a useful tool for performers. *Collaboration:* Although the system consists mostly in a graphical interface for the user, *spAAce* is designed taking into account audience perception and comprehension. In fact it is for live music purpose. *Learning curve:* It should be easy to start with spatializing sounds and give location to sounds, but it has a high ceiling keeping player to improve their skills. In fact the performer should learn how to manage the space dimension properly according to the music.

Context

Music style: The idea of our interactive system was born for an elitè of avant-garde and electro-acoustic music composer who started to play with the spatial dimension to convey music expressiveness almost 50 years ago. However with the spread of new spatial techniques, such as WFS and Ambisonic, the concept of spatial music is becoming more popular and applicable to many new electronics and experimental musical genres and movies soundtrack. *Physical environment:* our app is using a WFS techniques and require a WFS system, however new studies have been conducted to render spatial sound even with normal common headphones [1]. *Social environment:* the social context is wherever the spatialization of the sound is required, from a live performance in concert hall to a sonic installation, but also for mixing with digital audio workstations (DAW).

Specifications

- Control
- Mapping
- Operational freedom
- Embodied facilitation
- Feedback

4.3 Software and Components Design

Figure 4.2 show the first idea of our system. We designed our system according to the NIME model described in [27]. The overall system consist in many connected components, from the mixing console to the loudspeaker system which implements the WFS technique and to the controller for the users. However, developing such a huge system is a heavy task. Thus, we have decided to focus our attention on the *input device* and the application that models the position of the sound in the space, the *moving system*. This system follows the idea of the Model-View-Controller design pattern. We chose to exclude the mixing stage and work directly with given audio file, that can be provided and managed by a digital audio workstation (*virtual mixer*) o directly by the spatial render engine. Finally our system, controlled by the user, apply the spatialization of the sound sources (every *track* provided by the mixer) controlling the spatial parameter of the sound rendering engine. In some initial brainstorming, we decided to implement some spatial effects managements, however we postpone such implementation as future improvements.

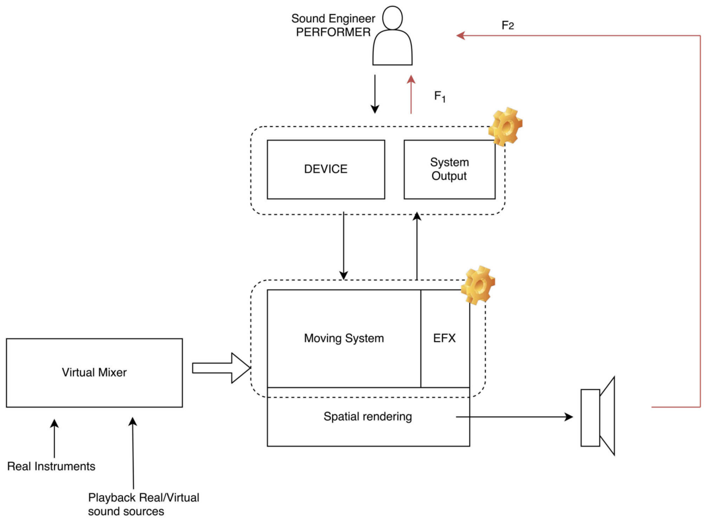

Figure 4.2: Sketch of the software architecture for the first *spAAce* prototype

4.4 Graphic Design

For the graphic design we started developing the application under the following constraints:

- The application should be as easy as possible for the performer.
- Follow the railway metaphor as much as possible.
- Enable the performer to manipulate sound sources properties fast enough to follow the performance in order to be able to recreate non real time performances.

The graphic design also takes concepts and ideas from other successful software editing programs such as paint, gimp, photoshop, or even video games such as Warcraft. Selecting a sound source will display its properties on the bottom right corner of the display, and multiple sound sources can be selected by using the selection panel under the bottom left corner of the display panel.

4.5 Interaction Design

Device Interaction

The device interaction is divided in two main parts. The mouse keyboard interaction and the leap motion interaction.

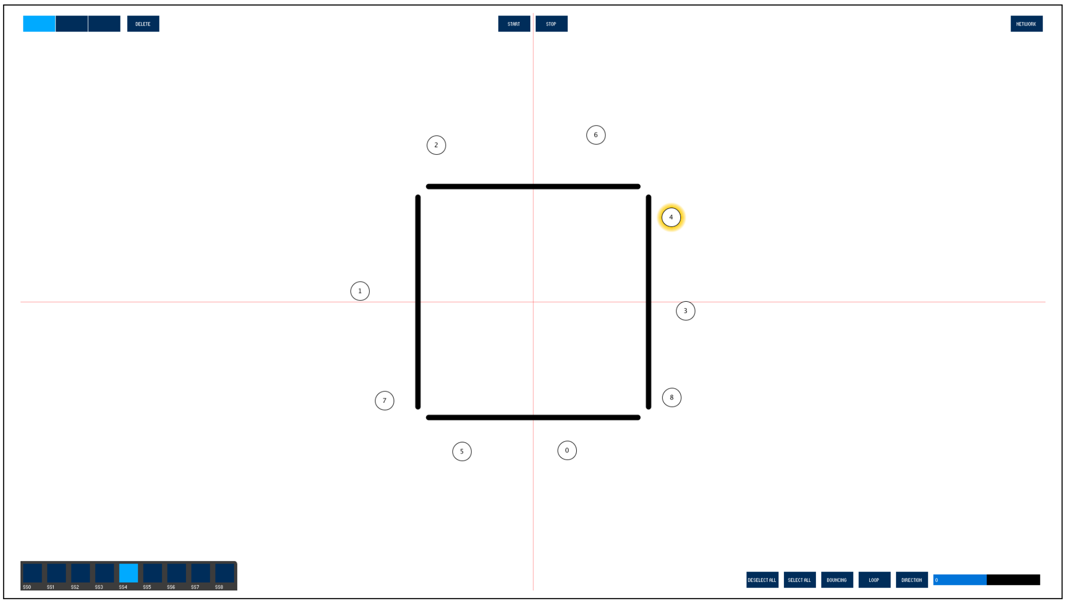

Figure 4.3: spAAce graphics example

Gestural design

For main gestures where implemented:

- **Pinch**: Used to apply an impulse to the sound sources. The user can use the pinch gesture to create a virtual spring that will apply that impulse to the sound source. The force will be calculated by the drag distance of the spring.
- **Grab**: This is use to create an attractor at the location of the hand. The z location is relevant so that the gesture will only be triggered when the user reaches some particular distance from the Leap motion controller.
- **Two handed gesture**: This is an experimental gesture used to control the velocity of the sound sources. It is done by drastically placing both hands over the controller and moving one hand further than the other.
- **Open hand**: Opposite gesture of the grab, the z location is also inverse, the user has to raise in order to trigger it.

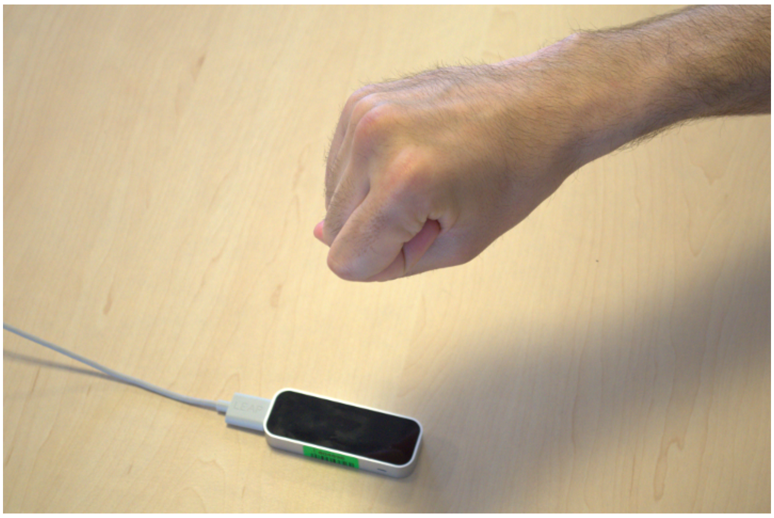

Figure 4.4: Grab gesture example

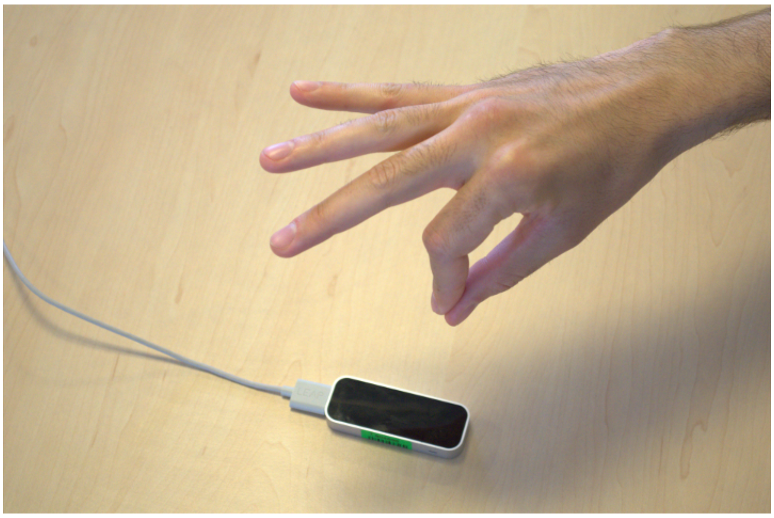

Figure 4.5: Pinch gesture example

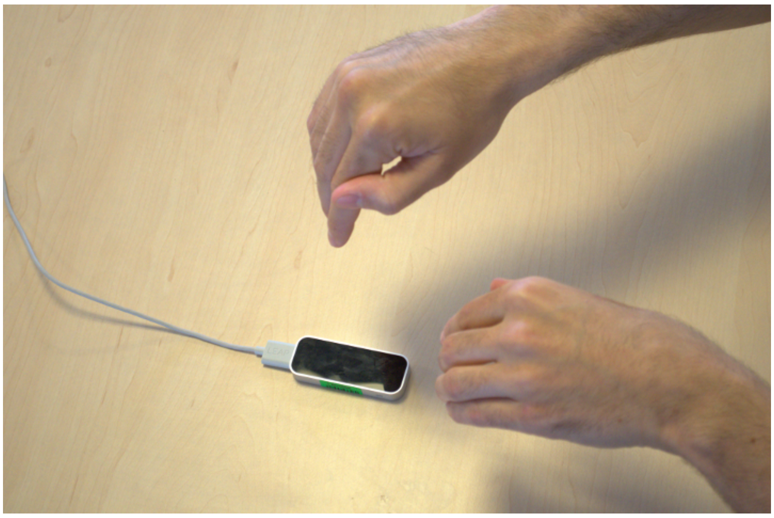

Figure 4.6: Two hands gesture example

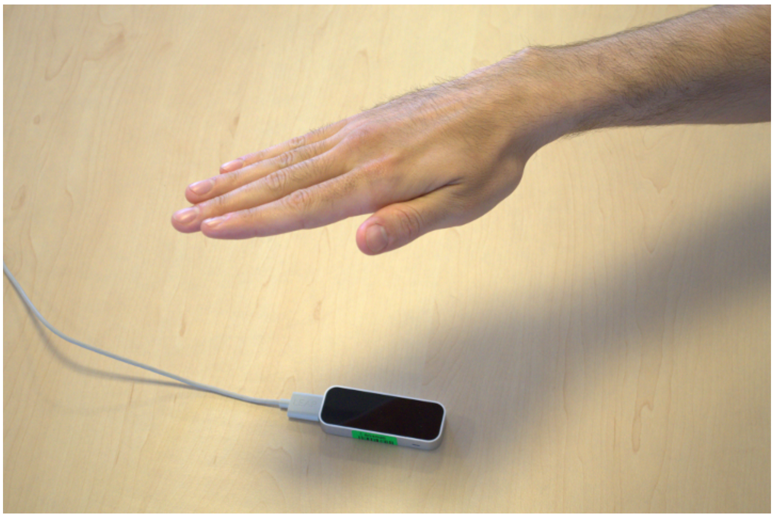

Figure 4.7: One hand gesture example

Chapter 5

Implementation

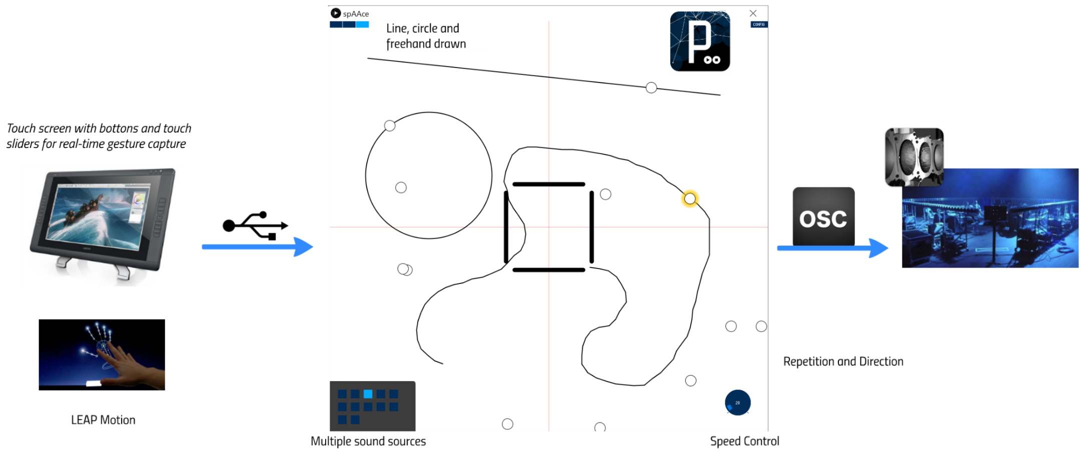

Figure 5.1: Current software architecture for the *spAAce* prototype

5.1 Softwate Architecture

As said in previous sections, the implementation for *spAAce* has been done in processing language. The software architecture is divided into four different main blocks. This chapter will consist of a more detailed analysis of these blocks, which are the following:

- Visualization
- Network
- Interaction
- Particle and Path Systems

The current implementation and software architecture is shown in Figure 5.1.

5.1.1 Visualization

The graphics have been done using processing system architecture. There are fundamentally two main methods to implement:

- **Setup:** It is used to initialize all the variable and processes. In our case, this method is used to initialize the gesture manager, create the interaction objects and controls, and instantiate the network classes.
- **Draw:** This loop is called when the drawing needs to be perform. The logic of this loop is designed in order to paint as in a canvas, what gets drawn on top is covering the layers below. Apart from drawing this loop is used to process the interaction events of the user by checking gesture actions.

```
void draw() {
  mainLayer.beginDraw();
  mainLayer.background(255);
  loudspeakersWFSsystem.displayCenter();
  loudspeakersWFSsystem.displaySystem();
  for (Trajectory t : trajectories) {
    t.display();
  }
  .
  .
  .
```

5.1.2 Network

The network subsystem is in charge of communicating the processing application with WFSCollider. The implementation is done by sending OSC messages via TCP directly to the WFSCollider application. Figure 5.2 shows how the *spAAce* class contains both *NetWorkPanel* and *OscMessageSender* objects as global references, but it is actually the *SoundSource* and the *ControlButtons* the one who use it when the methods are called. The code has been implemented under the **OscMessageSender.pde** class, and the messages required are the following:

- **oscName/1/point.x**: Set x position.
- **oscName/1/point.y**: Set y position.
- **oscName + prepareAndStart**: To start the composition.
- **oscName + stop**: To stop the composition.

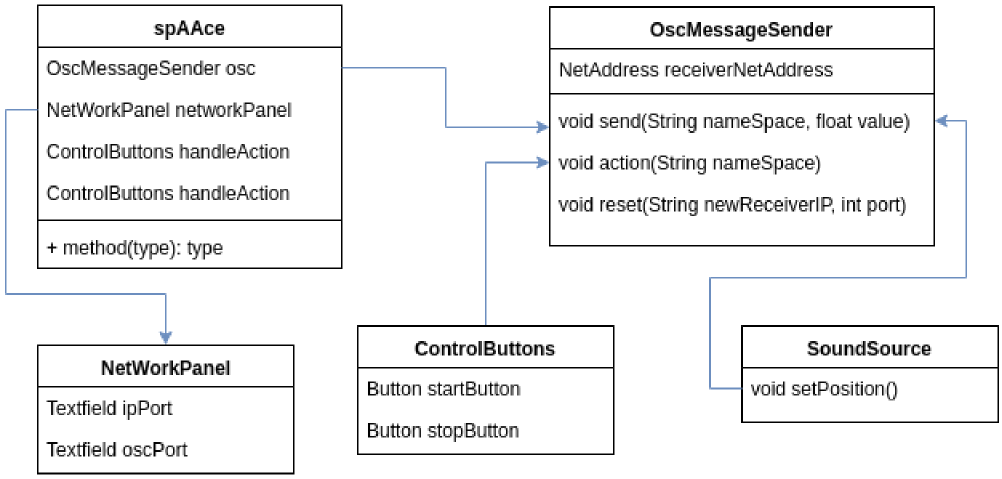

Figure 5.2: UML of the networking implemented in *spAAce* prototype

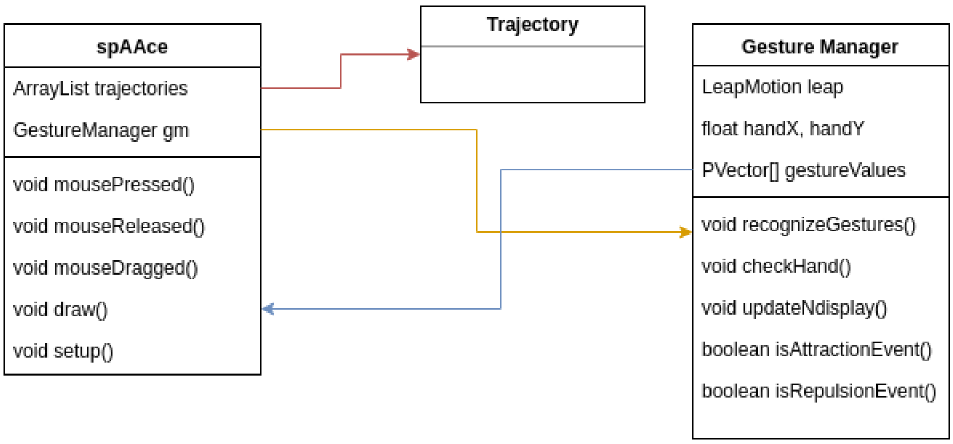

Figure 5.3: Gesture manager and spAAce interaction

5.1.3 Interaction

The interaction is divided between two main controllers, **keyboards + mouse** and **Leap Motion**. The implementation of the Leap Motion logic is done in the class *GestureManager.pde*. In this class the hands and gestures are recognised so that the rest of the application can perform different actions when detected. Figure 5.3 explains how the *spAAce* class contains an array of the created trajectories, but also a reference to the *GestureManager* class, and it is the draw method the responsible for updating the hand values stored in the *GestureManager* object. As explained in previous sections there are 3 main gestures the user can perform to interact with the sound sources.

- **Attraction event**: This gesture is recognised by asking the gesture manager whether the hand is closed and close to the leap.
- **Repulsion event**: Very similar to the attraction, it is recognised by detecting whether the hand is opened and far away from the Leap Motion.
- **Pinch event**

5.1.4 Particle and Path System

This is one of the fundamental parts of the system. In order to implement the path system, we followed the railway metaphor described in previous sections. For this reason two main classes were designed to interact with each other, the *Trajectory* class and the *SoundSource* class. The *Trajectory* class holds a list of sound sources which are currently traveling accross the path of that particular trajectory. The *SoundSource* class also holds a copy of the trajectory which it might be following or not, figure 5.4 explains the relationship between the *SoundSource* and *Trajectory* class. Each time the update method is called, the sound sources available are iterated and the method *StepInThePath* is called in order to move the sound sources across their paths. This method also checks whether the bouncing and/or loop properties are enabled to make the sound sources bounce or loop in the trajectory.

The **particle system** implementation has been achieved by giving physical properties to the sound sources as following:

```
float mass = 10;
float k = 0.1;
float damp = 0.95;
float force;
PVector position;
PVector velocity;
PVector acceleration;
int speed;
```

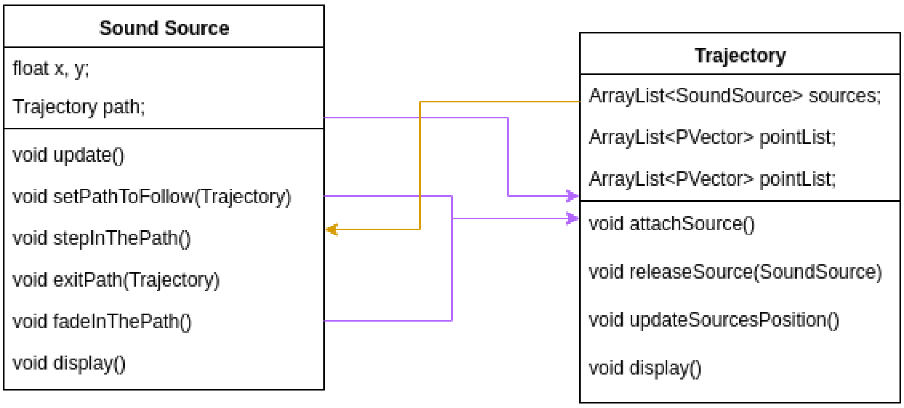

Figure 5.4: Description of Sound Source and Trajectory classes

```
float magAccel;
```

This allows the *Attractor* class to perform physical operations over the sound sources. The method applyForce is then used to make the sound sources move according to a particular impulse or force when gestures, for example, are applied.

```
void applyForce(PVector force) {
    if (isSpringEvent) {
      hasPath = false;
      acceleration.add(force.mult(-k).div(mass));
    } else {
      hasPath = false;
      acceleration.add(force.div(mass));
    }
}
```

5.2 Technical Implementation

Classes

Here is a list of the most important classes:

- spAAce (i.e. the Main)
- Trajectory
- SoundSource
- LineTrajectory (extends Trajectory)
- CircleTrajectory (extends Trajectory)

- FreehandTrajectory (extends Trajectory)
- GestureManager
- SoundSourcePanel
- SourceControlPanel
- Attractor (extends SoundSource)

External libraries

Three main libraries were used for the development of the processing application:

- **controlP5**: Graphical library for processing which allows to easily implement user controls such as sliders, knobs, buttons. this was mostly used for the velocity controls, trajectory selection and sound source selection.
- **oscP5**: This library makes it easier to implement the communication between the application and the WFS system by providing an interface such as *oscManager.send*.
- **de.voidplus.leapmotion**: Leap motion interaction library. It provides us with a *leap* object which itself offers the application with the method **leap.getHands**. This last method is the used to retrieve the single information from each hand.

Controller and Devices

The system is designed to work best with a Wacom surface tablet and a Leap Motion. The surface allows faster and easier interaction while moving the sound sources and assigning them to different trajectories. On the other hand the Leap Motion is not strictly necessary but it allows the performer to a more expressive and interactive environment while using gestural controls to manipulate the sound sources. For the connection with Wacom, standard VGA was used, along with a USB connector for pointer coordinate transmissions. For the installation of the LeapMotion controller refer to the guides in *https://www.leapmotion.com/*. The mapping between LeapMotion and screen coordinates was done empirically by trial and error, the parameters look something like the following, but they should not necessarily adapt to other screens:

```
PVector handPosition = hand.getPosition();
handX = map(handPosition.x, 200, 800, 0, width);
handY = map(handPosition.z, 20, 70, height, 0);
```

Chapter 6

Testing

It is common in the context of NIME or digital musical instruments (DMIs) to evaluate the first prototype to collect feedback from the user in order to improve it. In this chapter we will discuss our evaluation procedures.

To understand if *spAAce* is correctly abiding to our user's needs, we have tested the prototype of our system in two consecutive phase: an iterative phase followed by a final one. We have conducted some iterative tests during all the development and, finally, we set up a final and more accurate test at end of the entire process. The goals of the iterative testing were to gage product usability and evaluate the features we implemented, so that we could converge and focused only on the main ones. The goal of the final test was to provide a general and more complete evaluation of our system and show some concrete results.

Since the all project could be considered as prototyping the *spAAce* system and due the lack of an adequately evaluation framework for this kind of application and, in particular, for spatial sound artistic practice, we have decided to evaluate our system qualitatively. So we focus on capture the user's enjoyability related to the system usability.

6.1 The iterative testing

As mentioned in 4.2, in order to design our application we followed the framework proposed in [28]. We conducted some internal periodical tests on our classmate and friends in order to evaluate the usability of our system and the interaction we designed for it. Moreover, a constant meeting with our supervisors provided us to have more wise feedback .

6.2 Testing the *spAAce*

As the authors of [5] state, there are several ways to evaluate a NIME or a DMI: one approach is build upon the HCI research of evaluate input device for sound manipulation; another one is based on the qualitative testing and it aims to investigate on subjective qualities such as enjoyment, expressiveness and perceived *affordances*; others consider more the time-related issues, such as performer's practice and virtuosity. Since we are still in the prototyping stage, we decide to adopt the qualitative research instead a quantitative comparison. Thus we focused on investigating subjective qualities inherent to the musical experience, such as enjoyment, expressiveness and perceived affordances, both from the performer and the audience side. Figure 6.1 depicts the overall project evaluation framework.

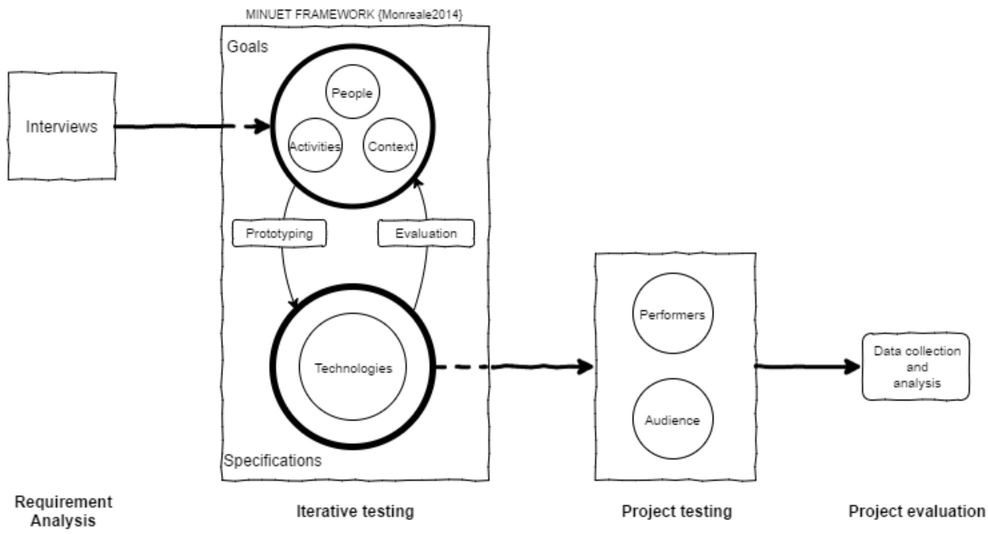

Figure 6.1: Block diagram of our evaluation framework for the overall project.

Although we had the conditions to conduct the *significance tests* to evaluate our project thesis, we have found during the early iterative testing that our system is so innovative and unconventional that the subjects satisfaction greatly influence the outcomes. Thus, since we could not avoid this, we thought that every test would be subjected by this great bias.

Subjects

Based on the previously described attempts and all our design process, we evaluated our system considering the *performer* as main beneficiary of our app. Thus, we have conducted some experiment related to the performer's perspective in a single-user qualitative experiment. This allowed us to get precious feedback and

Dimension space for the final evaluation

Subjects	*GUI testing*	*System expressiveness*	*Enjoyability*
Performers	Learnability Visual feedback (terminology and visualization)	Learnability/virtuosity Transparency of mapping (physical metaphors) Conducting vs. playing	User satisfaction
Audience	~	Transparency of mapping (cause, effect, mapping, intention, errors)	Audience satisfaction

Table 6.1: In this table the evaluation parameters are shown divided by typology and subjects

comments for future improvements and for the stakeholders' accounts. However, as stated in [5] and with much effort in [4], the audience's perspective cover an important role for evaluation a Digital Music Instruments (DMIs) and in particular a New Interface for Musical Expression (NIME) and it should be take into account. So that, our goal was to assess the participants comprehension abut five components of our system, which are explained in the following paragraphs.

In total we had nine participants for *performer* testing and sixteen for the *audience* testing. This participant number does not provide statically significant evidence, however it is a reasonable number of participants for evaluate the overall system prototype and retrieve some useful feedback and comments.

The Performers For the performers test we gathered nine voluntary students of Aalborg University. Seven of them were male, and 6 of them had previous musical experience. Four of them did not have any previous experience with similar software, and the rest reported to have some degree of experience. The age of the students was between **19 and 29** years old.

The Audience Sixteen graduated students from Aalborg University in Copenhagen voluntarily participated in the *audience* test. Participants in this study were studying a master degree in either sound and music computing or different fields of technical design, such as medialogy, and technological studies. The age of the participants was between **23 and 31** years old, and nine of them were male. Eleven of the participants had some previous musical background of some degree, and half of them had some degree of experience with similar software.

HCI testing for the GUI

The definition of usability is a reference to interactive multimedia software that has been discussed in [17] and in [23]. The author of the latter paper has established the five dimensions of usability that may be assessed in the user testing scenarios:

- Learnability
- Performance effectiveness

- Flexibility
- Error tolerance and system integrity
- User satisfaction

We decided to borrow this dimension space and adapt it to the context of the NIMEs or DMIs. And for the nature of a prototype we only focused on *user first reaction* such as a parameters for the learnability and the overall *user satisfaction*. We considered the *performance effectiveness* as parameter of the system expressiveness.

Information and terminology As pointed in [27] and summarized section 2.1, one of the key concept of designing this kind of systems is the **feedback**, that can be *primary*, that is related to the visual, auditory and tactile-kinestetic feedback, or *secondary* related to the sound produced by the instruments. Thus, as part of the usability testing, we evaluated the primary feedback, in particular the visual interface that should represent in real-time the state of the sound sources and should allow the user interactivity. In these sense we evaluated how much the our system keeps the user informed about what it is doing. We evaluate as well the terminology, that is the alphabet, we adopt to name the features, such as *bouncing* or *loop*.

Learnability In the HCI field, the *learnability* refers to the ease with which new or occasional users may accomplish certain tasks. In order to evaluate this aspect, we randomly divided the subjects in two groups: trained and non-trained. So that, only the person of the first group were asked to attend to a short tutorial in which the basic ways of interaction of our system has been presented.

Testing the *Expressiveness*

A good definition of expressiveness in the context of NIMEs and DMIs could be found in [9]. The author indicates as **expression** the "felicitous or vivid indication or depiction of mood or sentiment; the quality or fact of being expressive", and as **expressive** the "effectively conveying meaning or feeling". Within this context, many other authors proposed their own taxonomies and their criteria to evaluate NIMEs or DMIs, such as in [12] [30]. So we found and used the following common **dimension space**:

Learning curve and virtuosity All these authors state that instruments must have certain degree of complexity related yo the input control and the sonic results, to be considered potentially expressive. "So it is reasonable to expect that such an instrument will have a certain learning curve" [9]. So that, this instruments can be mastered by a performer who should achieve a level of virtuosity. This concepts has been extended, and the ideal new musical interfaces should have "low entry fee with no ceiling on virtuosity" [41].

Transparency of Mapping Since our system is a real-time multi-modal performance interface and since one of our goals is to *spice up* the act of spatializing sounds, our prototype should attempt to convey expressive communication for the user and the audience. Moreover "the expressiveness of an instrument is dependent on the transparency of the mapping for both the player and the audience" [9].

Thus, in this sense, the *transparency of mapping* in a key concept for what we wanted to evaluate from both the performer's and the audience's side.

Performer transparency As stated in [12], the factors that can make the mapping between instrument control, and the sound production psycho-physiologically transparent for the player, are the physical metaphors. And the possibility to learn to play on a reasonable human time scale. Thus, the transparency of a mapping for the users depends both on familiarity with effects of control parameters, and the level of dexterity the user has with the controls.".

Audience transparency The factors that can make the mapping transparent for the audience are more intuitively, and are related to the comprehension. In this study we used the dimension space suggested in [4]. The author measure the audience comprehension along the following aspects:

- Cause comprehension: "Which part of the performer's body (or which technological device) was used to interact with the system?"; "How understandable are the actions made by the user for interacting with the system?";
- Effect comprehension "Did the system provide enough audiovisual information for the audience to understand what is happening between the user and it?";
- Mapping comprehension "How clear is the relationship between the user's actions and the resulting sound?";
- Intention comprehension "How successful was the user to express himself using the system?"; "Was the user's intention well understood?";
- Error comprehension "Were the system's errors perceived (e.g. technical problems and software bugs)?"; "Were the user's errors noticeable?".

Conducting vs Playing The authors of [9] state that “it is important to [...] equate control with expression” and “music can convey meaning or feeling simply by its organization of sound elements, the performer of music provides expression by evidencing that organization, by adding shape and nuance to the given materials”. Thus, we decided to add a qualitative, subjective and provocative distinction for *mixing* and *playing* the sound sources, in order to evaluate if our system is perceived more as a mere mixing interface or more as a device that can enhance the *shape* and the *nuance*.

Testing the Enjoyability

The users satisfaction and enjoyability aims to articulate the users emotional responses to the system [25]. To evaluate this aspect we adopted the common qualitative approach, investigating subjective qualities inherent to the followings:

- Reaction to the application
- Performers enjoyability
- Audience enjoyability

The subjects were asked to evaluate the enjoyability of our system with some Likert-style scale.

6.3 Testing Environment

The experiments has been conducted the in the *Multi-Sensory* Laboratory of the Aalborg University in Copenhagen. We used the same setup described in 3.5.

6.4 Procedure and data collection

The tests were designed to provide feedback on the system usability and to gather general responses toward our product. According to the previous consideration, we have conducted two different test sessions: one for the users/performers and one for the audience (see Figure 6.2). In order to collect sufficient data from the participants we propose two surveys for each test asking qualitative questions[1] and we applied a 7-point Likert scale. We used such Likert scale since in [8] it has been reported that 10-point Likert scales produce significantly lower mean values than 5- or 7-point Likert scale.

[1] Attached you can find the surveys that we proposed to the participants

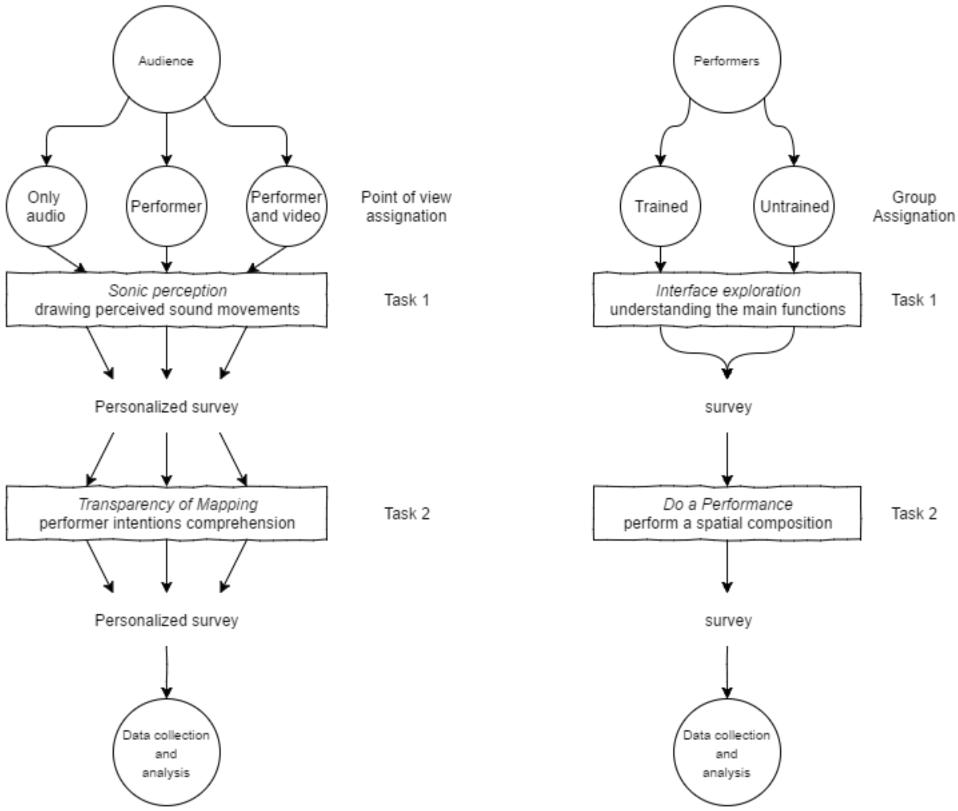

Figure 6.2: An schematic block diagram that summarize the procedure of testing our system on performers and audience

Testing the performers

To evaluate all the performer-related parameters, we designed two consecutive tests. The first one aims to evaluate the system and the interface exploration and learnability. For this test the participants were divided randomly in two groups, one with a small training, the other without. The groups were asked to perform some basic tasks to explore the main function of the system. Thus, assuming that all participants achieve the same basic knowledge after completing the first tasks, participants were asked to perform their own creative spatial composition as second and final part of the test. There were no time constraints for the testing and after the completion of the every tasks, users were then asked to fill out a *user evaluation* survey. For this testing four different tracks were prepared using both original compositions and stem tracks:

- Nine Inch Nails - Ghosts I, track 9.
- Nine Inch Nails - Ghosts I, track 1.

- Ludovico Einaudi - Night
- Original composition by Diego Di Carlo, 6.3

The users could choose one particular track of their own choice to improvise with.

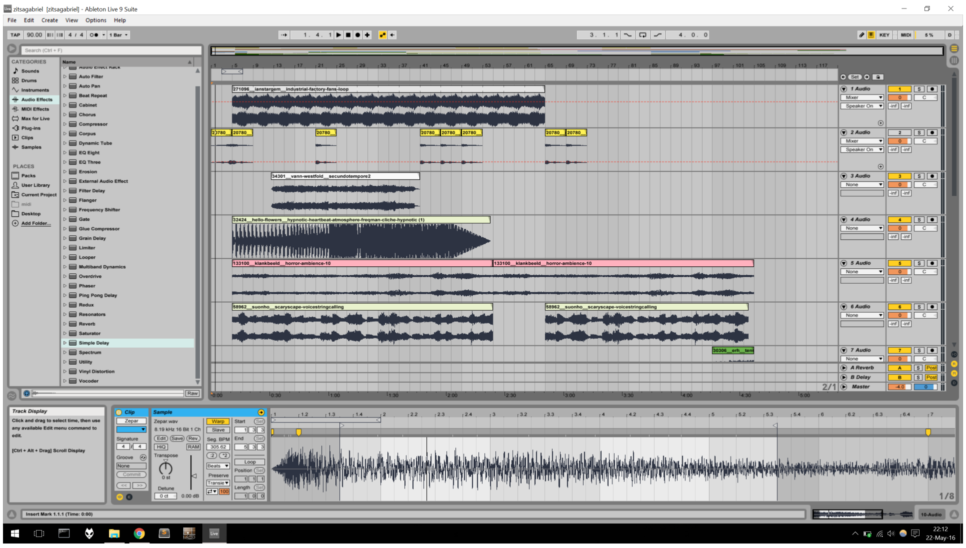

Figure 6.3: Original composition made in Ableton.

Disclaimer

Although the main hypothesis for this project is based around the idea that *spAAce* is supposed to be used by sound experts with a particular background, the performer tests were conducted by gathering non-trained participants. Even thought the results and discussion may provide some insight on how to further explore and develop the application, the tests should be repeated for trained sound participants.

Test 1 System exploration trained vs newbie

Test 2 During the second test, the participants were asked to create a composition of their own, by using the previous knowledge and by exploring, if possible, all the controls and features the application had to offer.

Testing the audience

In order to evaluate the audience comprehension of a performance, we designed two consecutive tests. Two different tracks were used for the testing, all of them downloaded in the form of stem tracks. These were remixed and adapted for the WFS system:

- Nine Inch Nails - Ghosts I, track 9, performed and spatialized by Matteo Girardi.
- Nine Inch Nails - Ghosts I, track 1, performed and spatialized by Diego Di Carlo.

For the first track, an simple spatialization performance was designed in order for the participants to be able to recognise shapes and patterns. The second one was designed in a more complex way, to test the understanding of the spatial composition.

Test 1 In this first test, the participants were placed in three different positions:

- **Blind**: The subject could not see neither the Wacom surface or the performance.
- **Partially transparent**: The subject could watch the performer. This is the normal circumstance of an venue or concert event, while ordinary people can not have access to the mixing stage.
- **Fully transparent**: The user subject watch both the performer and the Wacom surface display.

In this test, the users were asked to draw the perceived trajectories of the sound sources.

Test 2 For the second test, users were placed again in the same positions, but this time, they were told just to enjoy the performance and try to map the performance to the listening experience. The goal for this was to understand how the visual feedback affects both the transparency and mapping of the interaction of the performer with the system.

6.5 Statistics

In order to retrieve information and making inferences from the gathered data some basic descriptive statistics are choosen. All the collected data for features have been represent with thier first and second order statistics, that is *mean* and *variance*. Some basic data mining techniques, such as data visualization with charts and grouping using the speadsheet features, has been used further investigation. In order to prove the significance of some inferences and hypothesis testing, *t*-test and *Hotelling's T-square* test is used, both in the their independent samples fashion.

Hotelling's T-square test is the multivariate case for the classical and well-know t-test used to determine whether the population means of the two random variable are equal, i.e. $H_0 : \mu_x = \mu_y$. The definition of this statistic is

$$T^2 = (\overline{X} - \overline{Y}) \left[S \left(\frac{1}{n_x} + \frac{1}{n_i}] \right) \right]^{-1} (\overline{X} - \overline{Y}) \tag{6.1}$$

where S is the pooled sample covariance matrix of X and Y, namely

$$S = \frac{(n_x - 1)S_x + (n_y - 1)S_y}{(n_x - 1) + (n_y - 1)} \tag{6.2}$$

where S_X is the covariance matrix of the sample for X, $\overline{X}$ is the mean of the sample, and the sample for each random variable x_i in X has n_x elements, and similarly S_Y is the covariance matrix of the sample for Y, $\overline{X}$ is the mean of the sample, and the sample for each random variable y_i in Y has n_y elements [19].

Under the null hypothesis, it holds

$$F = \frac{n - k}{k(n - 1))} T^2 \tag{6.3}$$

where k is the number of variables. If $F > F_{crit}$ then we reject the null hypothesis with a level confidence α.

6.6 Results

In the following section the results are presented with respect to the dimension space and testing parameters presented in the previous section and summarized in the table 6.1.

Evaluating the GUI

We can divide the evaluation of the GUI in the following categories:

- Learnability
- Information
- Terminology

Learnability

According to the data measured, the GUI has revealed to be easy an intuitive to use. However there might be a problem with the data collection since not enought participants were tested and some of them knew the application on beforehand. Figure 6.4 shows the overall system experience regarding user usability.

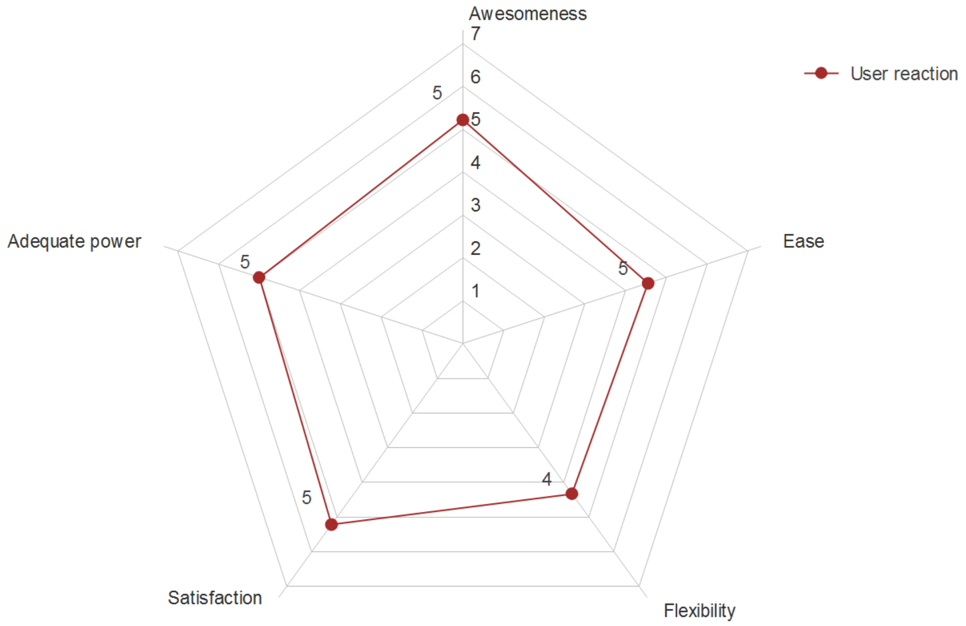

Figure 6.4: Radar chart that visualizes the mean of the Likert-type scale used in usability testing. The axis scale goes from 1 to 7 as the Likert scale used

Information

The mean of the data collected is almost 5.5 out of the 7 scale with a deviation of 1.58. The participants rated really well the GUI, strongly agreeing that it keeps the user informed and updated. There were some requests regarding audio feedback visualization which could be used to improve the GUI. Perhaps an implementation of editing trajectories as well as highlight or selection could be implemented for further improvements.

Terminology

The goal with this test was to measure whether the graphical interface terminology is related to what the user is actually doing. The data collected shows that this was optimal since the mean was 5.88 with a low deviation of 1.05.

Evaluating the performer experience

The evaluation of the performer experience is also divided into the following categories:

- Learning curve
- Transparency of mapping

- Leap Motion vs Wacom
- Conducting vs Playing
- Performance satisfaction and Enjoyability

Learning curve

Here we tested if the user-friendly interface and the natural approach of the Wacom tablet provides an easy and simple interaction. The answer is basically yes, but on the other hand, Leap motion and gestures were found difficult for every users (see Table 6.2) to manage and would probably need practice in order to be achieve better user control. Figures 6.5 and 6.6 show how the users rated the learning curve in the form. The main conclusion for this point is that the system has almost no entry-fee thanks to the already well known tablet interaction, designed and developed with a user-centered framework. Of course we have in mind the issues with the Leap Motion, but the research shows that training and improvement of gestures and mapping could lead to a mastering of the system.

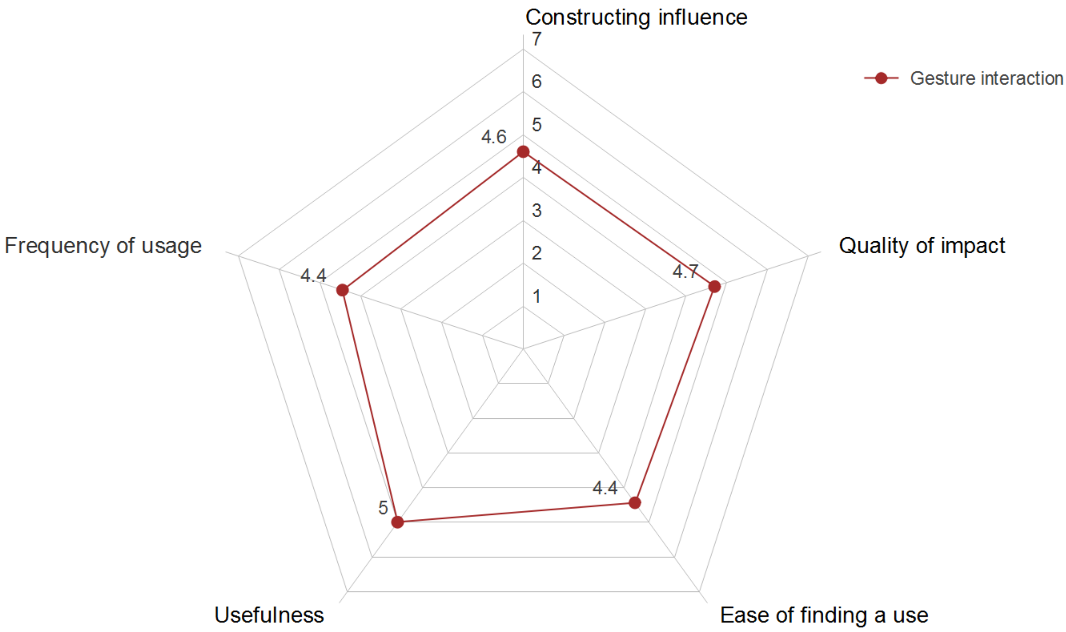

Figure 6.5: Radar chart that visualizes the mean of the Likert-type scale used for evaluating the gesture interaction between the performer and the system by the performer itself. The axis scale goes from 1 to 7 as the Likert scale used.

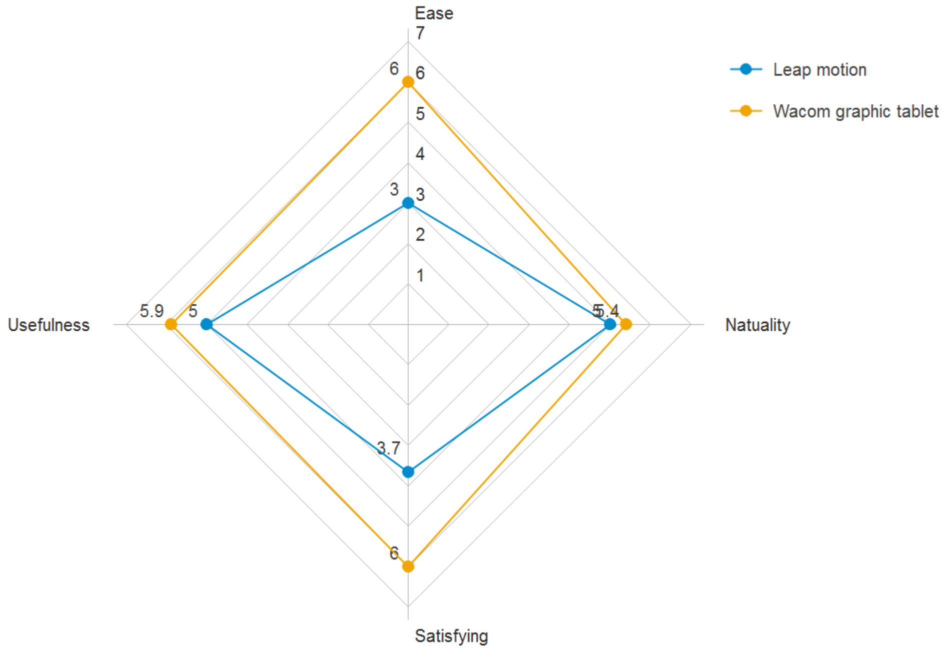

Figure 6.6: Radar chart that visualizes the mean of the Likert-type scale used for evaluating the input devices. The axis scale goes from 1 to 7 as the Likert scale used.

Transparency of Mapping

All of the functions offered by the GUI were explored by the users during the performance task. According to the users ratings and impressions we obtain similar results from the previous section, which are that the Leap Motion controls and gestures were difficult to execute. Figures 6.5 and 6.6 provide more insight into this results.

Performer reactions to gestural interaction

Variable and groups	Mean difference between groups					P-Value	Significance
	Terrible/Wonderful	*Difficult/Easy*	*Rigid/Flexible*	*Frustrating/Satisfying*	*In/adequate power*		
Musical background (have vs not have)	0.13	-1.13	0.73	0.2	-0.06	0.282	no
Knowledge about spAAce (have vs not have)	0.26	0.46	-0.66	0.06	1.26	0.390	no
Knowledge about WFS (have vs not have)	-0.06	1.4	0.6	1.46	-0.13	0.635	no

Table 6.2: Significance test using Hotelling's T-squared test statistcs for the gestural satisfaction parameters. *Null Hypothesis:* there is no significance difference between the mean vector of the two variables/groups.

Comparison between Leap Motion and Wacom table

Mean Difference between two controller evaluation parameters				P-Value	Significance
Difficult/Easy	*Un/natural*	*Frustrating/Satisfying*	*Useless/Useful*		
2.77	0.88	1.88	1.22	0.042	yes

Table 6.3: Significance test using Hotelling's T-squared test statistcs for the Leap Motion and Wacom Table reaction. *Null Hypothesis:* there is no significance difference between the mean vector of each couple of variables/groups.

Leap Motion vs Wacom

Again we find that there is indeed a significant difference between the tablet and the Leap Motion gestural interaction (see Table 6.3. In contrast the data and comments show that people find Leap Motion quite useful and natural in order to control sound sources, even if the experience was harsh and frustrating at first. Figure 6.6 supports this analysis.

Conducting vs Playing

According to the t-test *p*-value (0.36, alpha = 0.05) there is a significant difference between spatial music trained people and the ones who were not. The results show that participants thought that they were mixing rather than playing with the sound sources. We believe the title of the question was probably misunderstood. The "mixing" rated values belonged to participants highly trained in WFS techniques and/or amateur composers, while the participants who rated "playing" were less trained in these regards. There is perhaps an expectation bias and also a problem of terminology which could be improved in further tests.

Satisfaction and Enjoyability

The mean and standard deviation values plus the significance tests show that all the participants tested enjoyed very much the system performance, disregarding the background knowledge and familiarity with the interface (see Table 6.4). As show in figures 6.5, 6.6 and 6.4 the satisfaction was very high (5.88 mean plus 1.05 deviation for all the participants). After some discussion, we came to the conclusion that maybe there is an expectation bias for the novelty of the system.

Evaluating the audience experience

In this subsection we describe the results of the evaluation of the audience experience. The results are examined in the following categories:

- Sonic tasks

System enjoyment

Variable and groups	Mean difference between groups					P-Value	Significance
	Terrible/Wonderful	*Difficult/Easy*	*Rigid/Flexible*	*Frustrating/Satisfying*	*In/adequate power*		
Musical background (have vs not have)	-0.33	-0.93	-2.06	-0.26	-1.13	0.354	no
Knowledge about spAAce (have vs not have)	-0.86	1.86	0	0.73	0.06	0.390	no
Knowledge about WFS (have vs not have)	0.93	1.13	1.53	-0.33	1	0.109	no

Table 6.4: Significance test using Hotelling's T-squared test statistcs for the overall system satisfaction parameters. *Null Hypothesis:* there is no a significance difference between the mean vector of each couple of variables/groups.

- Transparency of Mapping
- Performance Enjoyability

Sonic Tasks

Although this was not part of our testing we decided to conduct it. The results show that participants were not able to draw trajectories properly and no quantitative evaluation can be done comparing the drawings made. Only 2 out of the 16 participants declared that they had no problems representing the moving sound sources in the space. The rest of the participants argued that they would probably need a tutorial in order to accurately represent the sounds, and even with this we highly doubt the representation could be better. [15] research shows that participants can be precise in this task while trying to represent single sound sources. Representing multiple suond sources can be a very hard task without previous knowledge. The data presented under 6.7 supports this conclusion, since the overall rate was 3.4 mean and 1.7 standard deviation with no significant difference on the Point of View.

Transparency of Mapping

According to the representation suggested in [4] and following their evaluation framework we found that the overall system perception is quite **transparent** to the user. Figure 6.8 shows that the fully transparent performance is one point above the partially transparent performance in the ratings. The data reveals that the multimodal interactive system designed is transparent disregarding the point of view of the user. The independent t-test and correlation analysis show that there are no evident bias related to the familiarity of the system, musicians or not musicians. The data proves that no information to support this has been found.

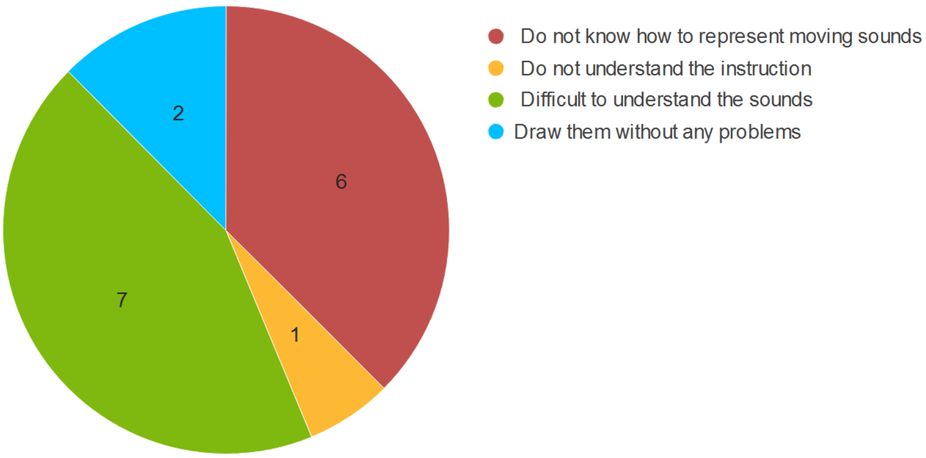

Figure 6.7: Radar chart that visualizes the mean of the Likert-type scale used for evaluating the audience transparency according to the evaluation framework describe in [4] the input devices. The axis scale goes from 1 to 7 as the Likert scale used.

Performance Enjoyability

The data reveals that all of the test participants enjoyed the performance. The ratings were 5.5 mean plus 0.96 standard deviation in the enjoyability scale. This strongly suggest that no significant difference between the three groups is to be found (p-value = 0.731 > 0.05). Our conclusion in this regard is that perhaps there is again some bias with the novelty of the system.

Some notes on transparency

A way to evaluate the transparency of our system is suggested in [13]. The author propose a two-axis plot, that is the audience transparency over the player transparency, form 0 (opaque) to 1 (fully transparent). In the Figure 6.9 we tried to represent the transparency of mapping according to this representation.

6.7 Test limitations and disclaimer

The system had some serious testing limitations regarding different aspects. The testing was performed only during two days, one for the audience and another one for the performers. Has described in the previous disclaimer, the performer participants were not composers but rather university students. Also some of the participants were perhaps biased for the fact of being friends and relatives. And finally as having been said in previous sections, the novelty of the system and the excitement of the new interface may have perhaps biased the subject evaluations.

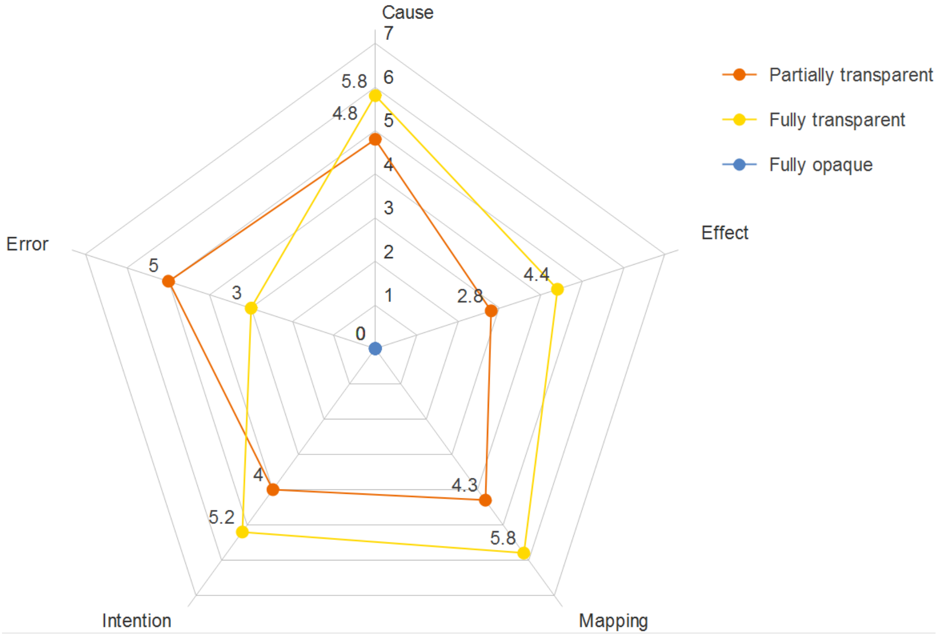

Figure 6.8: Subjects feedback for the sonic perception task, that is drawing the perceived path of the sound sources during a spatialized performance.

6.8 Discussion

spAAce is a sound spatialization tool developed primarily for the use within the artistic and sound engineer community. To ensure needs of the artistic and the technical needs are meet, the development process follows the MINUET user-centred design framework and has been iteratively tested to assess usability and provide user feedback. After 6 months of developing, we conducted a more concrete evaluation according the criteria of NIME and DMI community.

We tested our system mostly along three points of view:

- usability of the interface
- the expressiveness of the entire system and of the performance
- the enjoyment of both the user, that is the performer, and the audience

In terms of usability testing, the interface received very positive responses. No participants found difficulties in understanding all the basic features of the GUI. However, we know from for their feedback, that a number of improvements to the virtual space interface will be envisioned, with a particular focus on implementing some audio visual feedback and some common editing tools for the trajectories drawing system (such as copy, cut, paste and drag'n'drop).

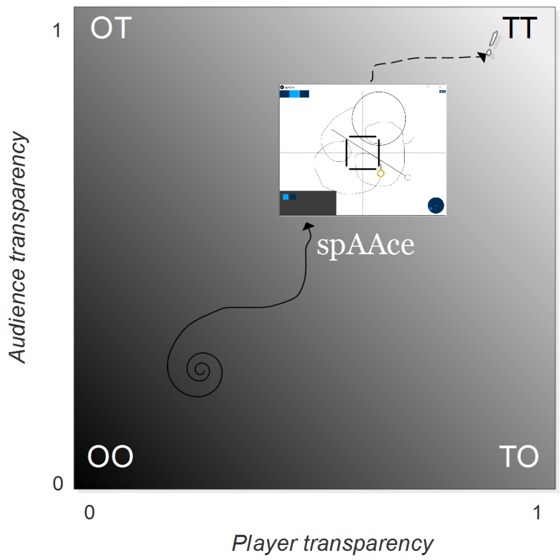

Figure 6.9: 2-axis representation of the transparency as define in [13]. By the considerations and the results given in this chapter, our system can be considered qualitative quite transparent for both the performer/player and the audience. Many further improvement will implemented in order to reach the top-right corner

The musical expressiveness of the system was evaluated expanding the terms along multiple dimensions, such as the transparency of the mapping, the effectiveness of gesture interaction, the learnability and the dichotomy of *conducting* vs. *playing*. Some subjects were asked to evaluate the system from the point of view of the primary user, that is the performer, while others from the audience perspective. All the dimensions have been evaluated by the subjects through quality rating. The outcome was positive: the overall system has been evaluated with an average of 5.55 out of 7. Both the performers and the audience have considered the system quite transparent: the interaction with the interface was found easy to understand and the physical metaphors of the features were appreciated. The same occurred with the gestural controls: subjects found natural and useful the gesture interaction, even it is difficult and a bit frustrating to handle. This shows that our system has a low entry-free, for the basic features, but it can be mastered, so that it can allow some level of virtuosity.

From the audience perspective, we tested marginally the sonic perception of the subjects and their understanding over the sound sources movements. Even if the overall reaction is very positive, almost the 90% of the subjects found difficulties in drawing the path of the sound sources. However this topic is related, but not crucial for the developing of the system.

A very positive feedback, ofter even passionate, was felt with respect of the subjects enjoyability and satisfaction. This shows that *spAAce* can be very enjoyable, even if the sound sources movements are not fully understood. However we think that the novelty of the system and the astonishment of watching the system for the first time working in the laboratory with an virtual acoustic environment, has influenced greatly the subjects.

Chapter 7

Conclusion

Since modern computers are becoming more and more powerful, it is now possible to have much more control on every single parameters required in a spatial music performance. In the recent years many different software have been developed to let composers and sound artists handle such complex situations in a more flexible and reliable manner. Indeed, audio spatialization is a relevant topic in many fields and providing a robust and efficient software is a goal for many researchers within the context of music spatialization. One of the main problems is that having so many different spatialization techniques, a software which satisfies all of them is hard to obtain and requires many years of testing and research. Moreover, the composers have different needs and requirements and for this reason it is even harder to provide appropriate tools. Furthermore, the equipment required for WFS or Ambisonics may be very expensive, then it is understandable that artists and performers choose to use simpler techniques for audio spatialization. In this paper we only focused on sound sources movements and gesture interaction and unfortunately many interesting ideas were discarded due to time constraint.

This report is indeed a first step to future developments and we will carry on improving the *spAAce* software. We hope that our effort in developing *spAAce* would be a useful starting point for future research in this field. Additionally, if *spAAce* is used in spatial music performances and live electronics, then our effort would surely be reward.

To conclude this report we would like to thank our supervisors, Dan Overholt and Francesco Grani, for providing priceless assistance while developing our semester project and Aalborg University to give us a comfortable environment where to study and to provide us the required equipments for our semester project.

Diego Di Carlo, Jorge Madrid Portillo and Matteo Girardi

Chapter 8

Future works

8.1 Software performance

Having an efficient software is absolutely essential to provide robustness and reliability while working with real-time applications, which requires low latency interaction. The performance measurements of *spAAce* were tested and provided a sufficient level of reliability, however several optimizations could be done. In order to achieve an optimal level of robustness, detailed tests must be carried out. There are several embedded limitations of the sound rendering engine (WFSCollider). One of the restrictions is the heavy computational load due to real-time process while moving many sound sources. Improving this limit would required hardcore programming of the sound rendering engine. The drawing routines are also not optimized and could be improved by using video rendering like OpenGL and similar. When it comes to interaction, the Wacom performs quite good with the tactile pencil. On the other hand, the Leap Motion is currently being updated to their latest software version, *Orion*, which would significantly improve the gestural controls and interaction.

8.2 Live performance of spatial music

Doing a live performance using *spAAce* would be a good evaluation test so as to understand its reliability, functionality and effectiveness. Hence, we are planning to do a concert in the Multi Sensory Lab at Aalborg Universty.

8.3 Gestures

Implementation of more gestures would surely benefit the overall reaction to the system and improving the gesture recognition software would contribute to achieve a better result. There are many possibilities and many ideas that come to our mind, nevertheless it is important to gather more knowledge and literature regarding such field to define a proper road-map for gesture implementation.

8.4 Other implementations

The implementation of visual feedback for the user, like for example, displaying when sound sources are producing sound, or somehow displaying the amplitude level of the sound sources could be a good improvement for the graphical application. Apart from this many ideas were discarded during the design process, such as adding different effects to the sound sources, like: filters, delays, reverberations, pitch shifting.... The parameters of these effects could then be manipulated using perhaps other gestural controls, or other forms of interaction.

8.5 Rendering Systems

This prototype system has been implemented together with WFSCollider which is indeed a powerful software for spatialization since it can be use with many loudspeakers setup such as stereo, quadriphonic and octophonic. However, avoiding our application to be bound with a particular software is surely an important feature to have. Therefore, extending the range of communication with other software and tools for spatial reproduction is absolutely a future improvement.

Bibliography

[1] Federico Avanzini, Luca Mion, and Simone Spagnol. "Personalized 3D sound rendering for content creation, delivery, and presentation". In: *NEM Summit 2009* (), pp. 12–16.

[2] Marije A.J Baalman. "Application of Wave Field Synthesis in the composition of electronic music". In: *International Computer Music Conference 2003, Singapore, 29 September - 4 October 2003*. 2003.

[3] Marije AJ Baalman. "Updates of the WONDER software interface for using Wave Field Synthesis". In: *LAC2005 Proceedings* (2005), p. 69.

[4] Jerônimo Barbosa et al. "Considering Audience's View Towards an Evaluation Methodology for Digital Musical Instruments." In: *NIME*. 2012.

[5] Jeronimo Barbosa et al. "What does" Evaluation" mean for the NIME community?" In: *NIME 2015-15th International Conference on New Interfaces for Musical Expression*. Louisiana State University. 2015, p. 6.

[6] Augustinus J Berkhout. "A holographic approach to acoustic control". In: *Journal of the audio engineering society* 36.12 (1988), pp. 977–995.

[7] *CodeOfLife WFSCollider*. `https://github.com/GameOfLife/WFSCollider`.

[8] John G Dawes. "Do data characteristics change according to the number of scale points used? An experiment using 5 point, 7 point and 10 point scales". In: *International journal of market research* 51.1 (2008).

[9] Christopher Dobrian and Daniel Koppelman. "The'E'in NIME: musical expression with new computer interfaces". In: *Proceedings of the 2006 conference on New interfaces for musical expression*. IRCAM—Centre Pompidou. 2006, pp. 277–282.

[10] Xavier Favory, Jérémie Garcia, and Jean Bresson. "Trajectoires: une application mobile pour le contrôle et l'écriture de la spatialisation sonore". In: *27ème conférence francophone sur l'Interaction Homme-Machine.* ACM. 2015, a5.

[11] Xavier Favory, Jérémie Garcia, and Jean Bresson. "Trajectoires: une application mobile pour le contrôle et l'écriture de la spatialisation sonore". In: *27ème conférence francophone sur l'Interaction Homme-Machine.* ACM. 2015, a5.

[12] S. Fels and M. Lyons. "NIME Primer: A Gentle Introduction to Creating New Interfaces for Musical Expression". In: *Workshop presentation at the New Interfaces for Musical Expression*. London. 2014.

[13] Sidney Fels, Ashley Gadd, and Axel Mulder. "Mapping transparency through metaphor: towards more expressive musical instruments". In: *Organised Sound* 7.02 (2002), pp. 109–126.

[14] M. Geier, T. Hohn, and S. Spors. "An Open Source C++ Framework for Multithreaded Realtime Multichannel Audio Applications". In: *Linux Audio Conference*. Stanford, USA, Apr. 2012. URL: `http://lac.linuxaudio.org/2012/download/lac2012_proceedings.pdf`.

[15] Amalia de Götzen et al. *The grouping of sound movements as expressive gestures*. 2013.

[16] Jim Highsmith. *Agile project management: creating innovative products*. Pearson Education, 2009.

[17] Simon Holland et al. *Music and human-computer interaction*. Springer, 2013.

[18] Tomlinson Holman. *Surround sound: up and running*. CRC Press, 2014.

[19] Harold Hotelling. "A GENERALIZED T TEST AND MEASURE OF MULTIVARIATE". In: (1951).

[20] Charles E. Ives. *Symphony No. 4*. Charles Ives Society Performance Edition, based on the Critical Edition, realized and edited by Thomas M. Brodhead. New York, NY: Associated Music Publisher, Inc. (BMI), 2011. URL: `http://www.musicsalesclassical.com/composer/work/47475` (visited on 05/25/2016).

[21] Sergi Jordà. "The reactable: Tabletop tangible interfaces for multithreaded musical performance". In: *Revista KEPES* 5.14 (2009), pp. 201–223.

[22] Jean-Marc Jot and Olivier Warusfel. "Spat: a spatial processor for musicians and sound engineers". In: *CIARM: International Conference on Acoustics and Musical Research*. 1995.

[23] Sung Heum Lee. "Usability testing for developing effective interactive multimedia software: Concepts, dimensions, and procedures". In: *Educational Technology & Society* 2.2 (1999), pp. 1–13.

[24] Trond Lossius, Pascal Baltazar, and Théo de la Hogue. *DBAP–distance-based amplitude panning*. Ann Arbor, MI: Michigan Publishing, University of Michigan Library, 2009.

[25] Adriana Madden. "Developing spaceJam: The New Sound Spatialization Tool for an Artist and Novice". Master of Music in Music Technology. MA thesis. Dr. R. Luke DuBois: New York University, July 2014.

[26] David G Malham and Anthony Myatt. "3-D sound spatialization using ambisonic techniques". In: *Computer music journal* 19.4 (1995), pp. 58–70.

[27] Eduardo Reck Miranda and Marcelo M Wanderley. *New digital musical instruments: control and interaction beyond the keyboard*. Vol. 21. AR Editions, Inc., 2006.

[28] Fabio Morreale, Antonella De Angeli, and Sile O'Modhrain. "Musical Interface Design: An Experience-oriented Framework." In: *NIME*. 2014, pp. 467–472.

[29] Sile O'modhrain. "A framework for the evaluation of digital musical instruments". In: *Computer Music Journal* 35.1 (2011), pp. 28–42.

[30] Dan Overholt. "The musical interface technology design space". In: *Organised Sound* 14.02 (2009), pp. 217–226.

[31] Rui Penha and J Oliveira. "Spatium, tools for sound spatialization". In: *Proceedings of the Sound and Music Computing Conference*. 2013.

[32] Rui Penha and J Oliveira. "Spatium, tools for sound spatialization". In: *Proceedings of the Sound and Music Computing Conference*. 2013.

[33] Ville Pulkki. "Virtual Sound Source Positioning Using Vector Base Amplitude Panning". In: *J. Audio Eng. Soc* 45.6 (1997), pp. 456–466. URL: `http://www.aes.org/e-lib/browse.cfm?elib=7853`.

[34] Francis Rumsey. *Spatial audio*. CRC Press, 2012.

[35] Stanley Sadie and John Tyrrell. *The New Grove Dictionary of Music and Musicians*. 3rd edition. Oxford University Press, 2001. ISBN: 978-01-951-7067-2.

[36] Jan C Schacher and Philippe Kocher. "Ambisonics spatialization tools for max/msp". In: *Omni* 500 (2006), p. 1.

[37] Gian-Marco Schmid. "Evaluation of musical instruments from the musician's perspective". In: (2015).

[38] Anne Sèdes, Pierre Guillot, and Eliott Paris. "The HOA library, review and prospects". In: *International Computer Music Conference | Sound and Music Computing*. 2014, pp. 855–860.

[39] Sascha Spors, Rudolf Rabenstein, and Jens Ahrens. "The theory of wave field synthesis revisited". In: *124th AES Convention*. 2008, pp. 17–20.

[40] Karlheinz Stockhausen. "Musik im Raum". In: *La Rassegna Musicale: Problemi della musica, oggi (Numero Speciale)*. Ed. by Guido M. Gatti. Vol. 31, IV. Giulio Einaudi editore, 1961, pp. 397–417.

[41] David Wessel and Matthew Wright. "Problems and prospects for intimate musical control of computers". In: *Computer Music Journal* 26.3 (2002), pp. 11–22.

[42] Karl H. Wörner and B. Hopkins. *Stockhausen: Life and Work*. [David Lewis Crosby collection]. ISBN: 978-05-200-3272-9.

Appendix A

Testing the audience

A.1 The survey

Refer to the attached PDF file.

A.2 The results

Refer to the attached XLS file.

Appendix B

Testing the performers

B.1 The survey

Refer to the attached PDF file.

B.2 The results

Refer to the attached XLS file.