RADBOUD UNIVERSITY NIJMEGEN

Hand Gestural Control of Sound: a Digital Musical Interface.

by

R.Barth

supervised by

dr. L. Vuurpijl and drs. A. Brandmeyer

A thesis submitted in partial fulfillment for the
degree of Master of Science in Artifical Intelligence

at the

Faculty of Social Sciences

Artificial Intelligence

August 2011

RADBOUD UNIVERSITY NIJMEGEN

Abstract

Faculty of Social Sciences
Artificial Intelligence

by R.Barth
supervised by
dr. L. Vuurpijl and drs. A. Brandmeyer

This thesis reports on the development of a hand gesture driven musical instrument. Using the Nintendo Wii remote controller in combination with IR-LEDs attached to a glove, users can draw gestures in the air. The system interprets these motions in two manners. First by classifying a repertoire of analytical control gestures. Second by deriving higher order motional features, also named as holistic control gestures. For the former a recognition performance of above 99% is reached with a single training example. For the latter, a music-motion study is conducted on listener associations between musical changes and hand motions. The results indicate that many motional features are significantly affected by many musical parameters. This provides essential knowledge for musical mappings utilizing the holistic gestures, which is investigated by a proof-of-concept prototype of the musical instrument.

"Your skills prove that you are a master artificer in your own right."

- Urza (Artifact Cycle 1:262)

Acknowledgements

I would like to thank dr. Louis Vuurpijl and drs. Alex Brandmeyer as primary advisors for their supportive help and great ideas during my internship. Louis, your knowledge on human-computer interaction and online handwriting recognition proved very valuable and helpful throughout the course of this project. Alex, your matching interests in electronic music, graphical arts and computational science were an ideal match to have on the team. Our meetings were very productive, though also engaging and most enjoyable. I would also like to thank dr. Rebecca Schaefer and prof. dr. ir. Peter Desain for their initial ideas and for their support to take this project abroad. Furthermore, I'm grateful for the insightful views of dr. Makiko Sadakata and again dr.Rebecca Schaefer on my experimental designs. Also my thanks go to drs. Rutger Vlek for his great musical ideas and affection to the project. Lastly I must not forget to thank Gerard van Oijen for his great efforts in creating such wonderful hardware for the glove.

Contents

Chapter 1

Introduction

Traditional musical instruments have gradually been complemented with electronic counterparts. The advent of computerized virtual instruments introduced a whole new genre where more often than not, music is composed through computer keyboards. The Stanford and Princeton Laptop Orchestras are prominent examples of this digitally created live music [19, 53]. Concurrently gestural control has established itself as an interaction paradigm, enabling novel forms of rich user interactions. Digital devices are no longer solely controlled by mouse or keyboards but recognize complex repertoires of multi-touch gestures. Moreover, the use of 3D gestures has entered our living rooms through the use of popular control devices like the Microsoft Kinect, Sony Playstation Move, and the Nintendo Wii. These developments on musical and gestural control provide the setting for the work presented here. This thesis reports on the research and development towards an hand gestural controlled digital musical instrument, which combines both wonders in digital music making and computerized understanding of physical gestures.

The instrument is designed to provide an intuitive and natural form of interaction by recognizing musical hand gestures in mid air. For this purpose an holistic interaction paradigm was adhered to. This type of interaction affords the user an unnoticeable direct transition between actions and sounds [25]. This in contrast to analytic systems, where the attention of users is directed towards analyzing their actions. In the context of music, the holistic approach is reflected by a phrase by Marc Leman:

> *"What is needed is a transparent mediation technology that relates musical involvement directly to sound energy. Transparent technology should thereby give a feeling of non-mediation, a feeling that the mediation technology "disappears" when it is used."* - Leman [35]

It is such an interface we strived for by combining an selection of pre-defined hard- and software technologies. This interface is used as a basis for a hand gesture driven musical instrument (DMI). The system detects and transforms the two dimensional hand motions to musical control, which is analogue to buttons or switches in regular instruments. Further visual feedback of the user's gestures is provided. Two types of gestures are distinguished by the system: analytic and holistic musical control gestures. The former acts as discrete control for musical events, such as *play* or next setting, whereas the latter is the primary musical controller for continuous sound production.

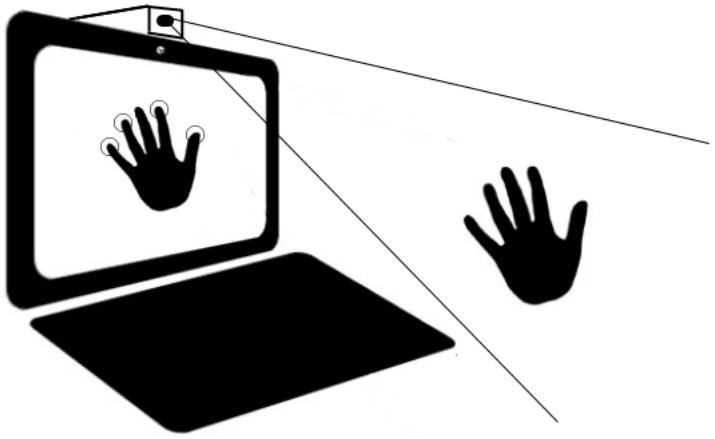


FIGURE 1.1: A schematic impression of the instrument in the design phase. A camera detects the hand and a computer system provides visual feedback.

Designing an holistic gestural DMI requires that the system complements the physical capabilities of the user and interprets the gestural message conveyed. Musical involvement is often based on corporeal articulations [5, 11] and captures which idea that created sound structure encodes aspects of the user's bio mechanical energy from actions. The theory of embodied music cognition describes this relationship between a human subject and its environment, analyzing the coupling of action and perception along with the body's engagement with music [35]. Until now, the design of previous input devices and their interaction techniques has been driven more by what is technologically feasible than from an understanding of human performance [9]. To design more usable interaction techniques, a more user centered gestural design should be embraced. This implies that research to gestures and their relation to music is equally important. Therefore this thesis reports on a study conducted of user's associations between musical parameters and motional features. The knowledge thereof will be utilized to shape intuitive musical mappings in the DMI.

The following research questions are visited in this thesis.

> R_1 *Can we design an affordable, portable interface that captures hand/finger gestures in an accurate, fluent manner without delay?*
>
> R_2 *Do there exist natural/intuitive repertoires of analytic and holistic hand/finger gestures?*
>
> $R_{2.1}$ *Do these gestures adhere to well-known usability constraints such that they are easy to learn, easy to use and distinguishable by the system in a robust and efficient manner?*
>
> R_3 *Do there exist relations/associations between sound and holistic gestures?*
>
> $R_{3.1}$ *How can we use this information in a musical performance?*

Organization of the the thesis In Chapter 2 the field of musical gestures and gesture driven musical instruments is explored in order to discover how it shapes the design of our interface. In the following chapter the development of this interface, both soft- and hardware, is presented and results in the description of the basis of the hand gesture driven instrument. In Chapter 4 the development of a repertoire of analytic control gestures is created. Furthermore a classifier system is evaluated on the recognition of this repertoire. Chapter 5 presents a study on listeners associations between musical parameters and hand motions. The knowledge thereof can be used as a basis for musical mappings from motions to sound, as described in Chapter 6. This chapter also reports on a developed musical prototype. In the final Chapter 7 our findings are concluded.

Chapter 2

Gesture Driven Digital Musical Instruments

In this chapter the development from regular musical instruments to electronic variants is explored. First it is described how the invention of electronic musical instruments (EMIs) evolves to digital musical instruments (DMIs). Subsequently it is investigated what the definition of gestures constitutes and entails, whereafter different types of musical gestures is explored. At the end of the chapter, the combination of gestures and DMIs is covered in order to discover how it shapes the design of the hand gesture driven interface.

2.1 EMI & DMI Development

Since the 18th century, musical instruments have made use of electricity. The first electrified musical instrument was the 'Denis d'or', invented in 1753 [13]. Strings of a piano were electrified to enhance the sound they produced. However the sound output was not amplified until in 1861, when the first speaker was created by Johann Philipp Reis. In 1876 the first electronic musical instrument was developed: an electric synthesizer, invented by Elisha Gray [10]. Sound was controlled by a vibrating electromagnetic circuit which resulted in the underlying concept of an oscillator.

An electronic musical instrument can be defined as a musical instrument that generates sounds by utilizing electric power. Further, the instrument outputs these generated sounds as an electrical audio signal amplified by loudspeakers. EMIs have a direct electronic relationship with sound output. This in contrast to DMIs where a microprocessor mediates the output by altering a digital representation of sound. Therefore,

DMIs represent a subset of EMIs. Before the beginning of the 21st century, EMIs were primarily designed as an output device, with synthesizers as primary sound generating devices. In 1954, Max Matthews developed the first sound generation program at Bell Labs. After personal computers made their affordable entrance into homes and offices, musicians became proficient in utilizing this new computer potential. At first only used as a sequencer or sound editor, soon synthesizers could be emulated virtually matching the same quality as their hardware equivalents. However, DMIs were still largely built like synthesizers with the desire to be controlled by keyboard-like inputs. It became common practice that DMIs largely powered these virtual synthesizers while still using the keyboard paradigm [45].

The field of DMIs is currently very active. Since 2001 the conference of New Interfaces for Musical Expression (NIME) [2] is annually organized, drawing researchers and musicians to share their developments of new technologies for musical expression and artistic performance. The rise of research institutes like CCRMA, IRCAM and MIT Media Lab show the desire to gather further knowledge around DMI's. Numerous state of the art DMI examples are also occasionally presented at the The Music Hack Day series [1]. In Figure 2.1 Berkeley University professor David Wessel plays on one of his custom DMIs: the SLABS. The instrument consists out of an matrix of pressure sensitive touch pads, capable of sending finger coordinate and pressure data to a computer. The software MAX/MSP [3] then translates these events to sounds.

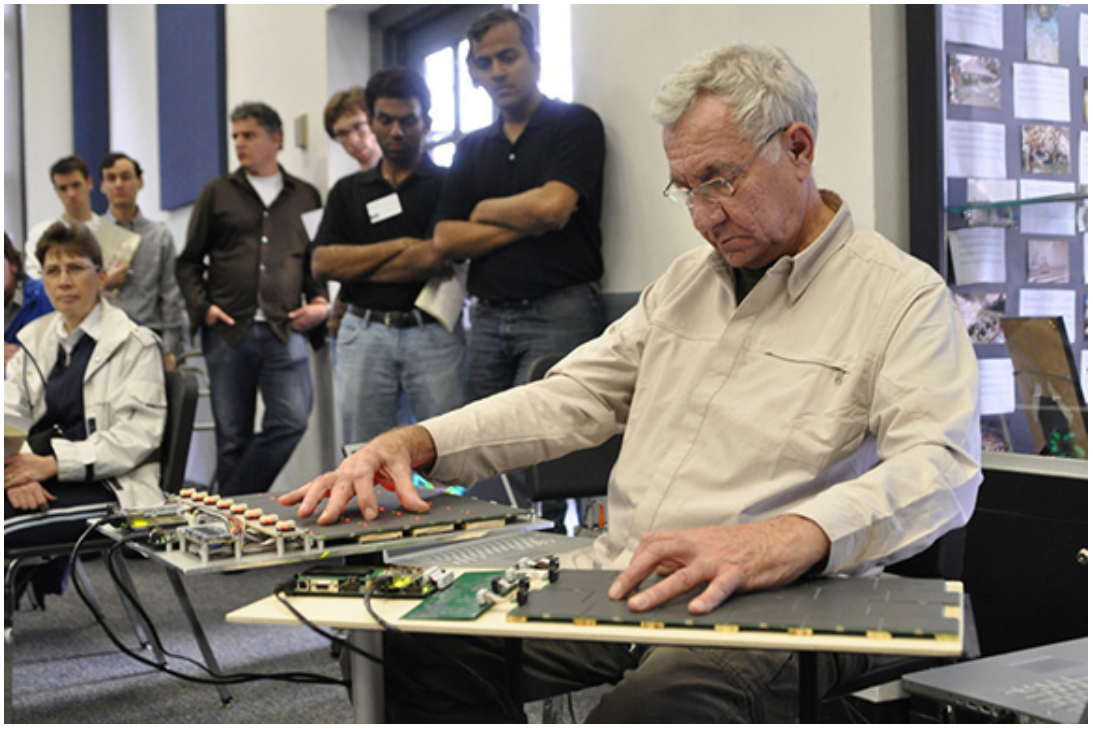


FIGURE 2.1: The SLABS being played by Berkeley University professor David Wessel.

During the course of this thesis, the following definition of a DMI is adhered to. It entails that in DMI development, both hard- and software are equally important.

> Def: *A digital musical instrument is a device where a microprocessor controlled device mediates between hardware input and audio output by processing/transforming a digital representation of sound. This requires that the computer does not solely act as a direct coupler of hardware input to audio output but processes the input to a higher level of information.*

2.2 Gestures

Wherever music is present, movements are ubiquitous. Literally by the vibrations of the air, but furthermore by the people moving to the sounds they perceive or make. People surrounded by sound often dance, wave or imitate the source of the sound [5, 56]. The movements that accompany sounds are coined as 'musical gestures' [28]. The term gesture has a broad range of definitions and refers to a great variety of phenomena. A general definition of a gesture is given by Hatten [24]:

> Def: *"A significant energetic shaping through time."*

However this broad definition can have too many interpretations. The physical characteristics as a vehicle of information should be added to the definition to emphasize the usage of gestures by humans:

> Def: *"A significant bodily motion through time, bearing meaning."*

A comprehensive framework for the categorization of gestures can be made using the work of McNeill and Zhao [28, 42, 61]. Three categories can be distinguished: communication, control and metaphor.

Communication Gestures The fields of linguistics, behavioral psychology and social anthropology primarily make use of the term communication gestures. These gestures convey information in social interactions. Examples are the physical movements accompanied with speech, like hand gestures and facial expressions or even movements which generate speech or writing. These communicative movements are also named gesticulation. They are not accidental irrelevant motions, as McNeill [41, 43] showed that these gestures contain communicative information.

Control Gestures Human-computer interaction (HCI) is interested in how gestures can be used as an input for controlling computers. Traditionally, humans have only partially interacted with computers by using gestures. For example, the entire gesture involved pressing a key on the keyboard cannot be seen as a significant gesture since the movement as a whole holds no inherent information. More recently HCI is trying to expand this interaction by recognizing more complex hand gestures [14] or body gestures [50].

Metaphoric Gestures Instead in the physical domain, gestures can further be viewed metaphorically. The term is best explained by an example of Middleton [44] who writes: *"How we feel and how we understand musical sounds is organized through processual shapes which seem to be analogous to physical gestures."*. Hence a gesture is here defined as a sensational interpretation as a metaphor for a physical event.

2.2.1 Musical Gestures

Musical gestures are gestures with any relation to music. Based on the works of Jensenius *et al* [28] there are four main categories of these gestures discernible:

- **Sound-producing gestures.**
  Gestures directly generating sound either by *direct excitation* or *modification*. Striking a string in a guitar is excitatory whereas bending the guitar's tremolo/vibrato arm, thereby creating a vibrato or a portamento effect, is a modifying sound producing gesture.
- **Communicative gestures.**
  Gestures which serve the main purpose of communication either *performer-performer*, *performer-perceiver* or *perceiver-performer*. For example in a musical ensemble a conductor indicates tempo with perceiver-performer communicative gestures, trying to control the sound production. The term controller-performer is also appropriate here.
- **Sound-facilitating gestures.**
  Gestures which support the sound-producing gestures, but are not directly involved in the production of sound. For example in piano playing, the movements the hands, arms and body make *in addition* to the fingers which hit the piano keys.
- **Sound-accompanying gestures.**
  Gestures which do not produce sound, but accompany or follow the music as a reaction to them.

Note that a specific musical gesture can fall into multiple categories. For example, a sound-accompanying gesture can also be communicative.

Based on these distinct types, a musical gesture can be defined by extending the general gesture definition in the following way:

> Def: *"A significant bodily motion through time, bearing meaning, that goes along with music, either while producing, adjusting, communicating, facilitating or accompanying the music."*

2.2.2 Analytic and Holistic Musical Control Gestures

The interface we created is designed to work similarly to an instrument. Therefore sound needs to be controlled. Since the interface further is hand gesture driven, musical controlling gestures need to be recognized. We can distinguish two types of these musical control gestures, inspired from writings of Marc Leman [35].

- **Analytic Control Gestures.**
  Motion information is not used during the gesture, only the resulting motion symbol counts. Similar to pressing buttons. Analytic refers to discrete and rational decision making, like for instance in a classifier. They can also be described as discrete control gestures. The analytical gestures have a binary and thus discrete existence, one gesture is either present or it is not.
- **Holistic Control Gestures.**
  Motion information is continuously used. A change in motion results in a direct change of the control of sound. Holistic refers to a higher level of reasoning or processing, where the actual motion pattern is not of prime interest, but for example its inherent features or attributes are. They can also be described as continuous control gestures. The holistic gestures are more continuous in the sense that at each point in time inherent properties of the gesture are of interest.

The analytic gestures will be used in the system for event control and not for direct music production. An example would be switching between instruments or quitting the program. The holistic gestures will be used for direct sound production and manipulation.

2.3 Hand-Gesture Driven Digital Musical Instruments

A subset of digital musical instruments require hand gestures as input. Because our interface strives for *'making music in the air'*, the focus in this section will be on hand gestures made without physical interaction with objects, like keys or strings as in regular instruments. Also named as remote hand tracking.

Using hands to create gestures for sounds as opposed to other body parts is not unreasonable. Hands are the main parts of the body used in manipulating the environment and have a wide degree of movement and positioning freedom. Hand gestures are a combination of rough torso, less rough arm, fine wrist and detailed finger movements. Hence the positioning capability for creating gestures is large. Further, the most common instruments are controlled with the hands [45], hence their proficiency in creating gestures for music is already proven successful.

Below a summary is given of sensing techniques that can achieve remote hand recognition and tracking. In the next section it is described which combination of these techniques is used for our interface.

- **Electromagnetic Sensing**
  This utilizes the interactions between magnetic fields of different objects. Antennae creating such a field can be used as a sensor detecting moving hands.
- **Optical Sensing**
  Cameras output consecutive frames which can be analyzed to distinguish a 2D[60] or 3D[34] hand motion. LED markers can be worn [7] to facilitate the recognition.
- **Acoustic Sensing**
  A high frequency sound source, typically 20-40 kHz, is tracked by 3 orthogonally placed microphones [58]. Tracking is achieved by taking into account the time the sound takes to arrive at each microphone. The disadvantage of such systems is that only a small space can be used for tracking ($<$1 m^3) and only one source can be tracked each time instance. It is also sensitive to differences in air temperature and humidity, wind, occlusion, ultrasonic noise and echoes.
- **Inertial Sensing**
  Relative hand position is determined with an accelerometer and a gyroscope attached to the hand. The sensors provide information of direction and the speed differences. Most sensors are slightly imprecise, resulting in inaccurate acceleration reports and thus positional drift. For example, a bias of just 1 milli-g (0.0098 m/s^2) results in a drift of 4.5 meters over 30 seconds [58].

Often multiple techniques are combined to overcome individual disadvantages or to increase the tracking accuracy. For example, a bias inertial sensing can be corrected with optical sensing information.

Examples The most prominent and one of the earliest examples of electromagnetic sensing for sound control is the theremin/aetherphone [20], ceated by Professor Léon Theremin in 1919. The main design consists out of two antennae. Both sense the positions of the user's hands and converts this information to an electric signal by controlling either the frequency or amplitude of an oscillator. This electric signal is then amplified and transformed to sound by a loudspeaker. This EMI does not require physical interaction and can 'track' the hands in mid air.

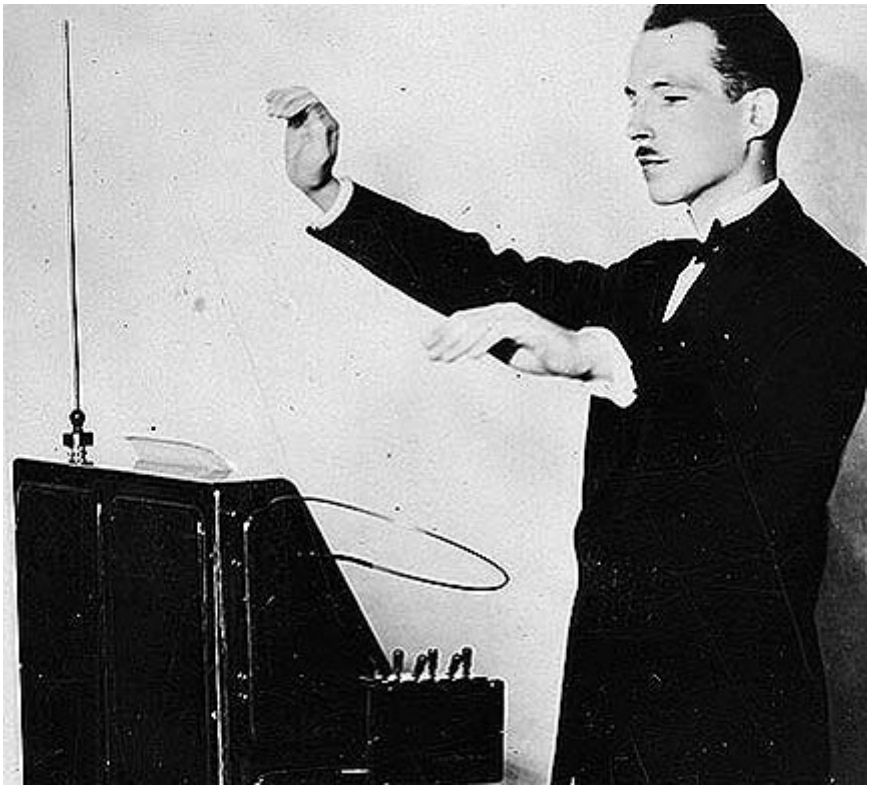


FIGURE 2.2: A theremin being played by its inventor Léon Theremin.

In later years, a wave of conductor-following systems were created, tracking hand motions indirectly by following the baton[1] of a conductor. In 1983, Haflich and Burns [23] used an combination of acoustic and optical sensing in order to track a baton in two dimensions. It was the first system to extract and analyze the conductor's gestures[2]. A later system

[1] A stick that is used by conductors to exaggerate and enhance the hand movements with the purpose of directing an ensemble.

[2] Previous research of conductor's gestures used joysticks, knobs or tablets to derive motions [39].

by Max Mattews made use of a baton emitting radio frequency signals which were detected by a metal plate [40] (see Figure 2.5).

Acceleration sensing was used in 1989 in the MIDI Baton, developed by Keane and Gross [29]. Changes in acceleration cause contact between a metal ball and the baton, triggering an electrical signal. The main purpose of this system was to detect beats. Positional data was not obtained. Another, more recent system used the Nintendo Wii remote controller's accelerometer to recognize hand gestures [49]. Again no positional data was obtained, just relative differences in direction. Optical sensing was used in another baton device, using a CCD camera in 1992 [7]. A lamp on the baton's tip was placed which was tracked by software that read out the CCD camera's data.

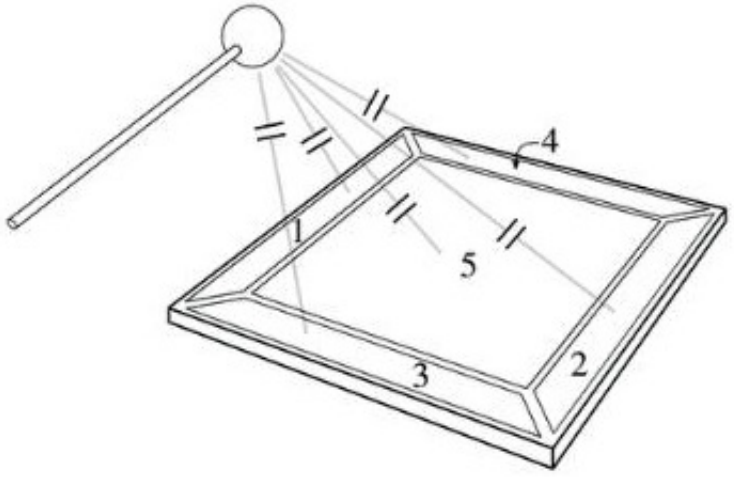


FIGURE 2.3: The radiobaton. A metal plate detects radio waves in order to track the baton.

In 1997, a more sophisticated baton was created, combining multiple sensing technologies. The Digital Baton by Marrin and Paradiso [38] contained an infrared LED at the tip of the Baton, a pressure sensor and acceleration sensors. The LED was tracked by a camera and the other sensors provided additional gesture information.

These baton-type of systems are still being developed today. Recently, Sony Computer Entertainment released the PlayStation Move: a motion-sensing game controller platform for the PlayStation 3 game console (see Figure 2.4). The working principle is also based on optical sensing, although more sophisticated than the Digital Baton or Bertini's Baton. The controller uses a sphere to diffuse RGB-LEDs light. The resulting light blob is then tracked as a marker by the PlayStation Eye, a plugin webcam for the gaming console. The system automatically derives the most distinct color in the surrounding scene and applies this to the controller to be emitted. The color is dynamically updated such that the tracking is optimized.

FIGURE 2.4: Sony's motion-sensing controller: the Move. The left semi-translucent sphere acts as a light diffuser.

Other Interfaces Some interfaces which are not designed particularly for hand gestural musical control, have the potential to do so.

One of such systems is the Color Glove, created by Wang [57]. It is capable of accurate and fast tracking the position and posture of a glove with a color pattern. Wang suggests that it could be used in in artistic musical applications.

FIGURE 2.5: The color gloves. Colored areas on the glove fortify recognizability by optical sensors.

Another system developed by Johnny Chung Lee tracks fingertips by using the Nintendo Wii remote gaming controller's infrared camera. The system can track infrared reflections from fingertips [32, 33]. The Wii controller is capable of tracking up to four blobs of infrared light and transmits this information wirelessly to a computer via Bluetooth. This results in an accurate 2D hand motion tracking system with little delay and a relatively high refresh rate (μ accuracy: 1mm, μ delay: 49.6 ms, μ refresh rate: 98 Hz [31]).

Another advanced motion tracking system is Microsoft's Kinect [50]. Apart from full body motion tracking it can recognize individual body part positions. A camera plus depth sensor outputs via software a 20-joint representation of the user's body. Recently, it was made accessible through the release of a non-commercial development kit[3]. The main downside of this system is the low refresh rate (μ 30Hz). Further, the resolution

[3]http://research.microsoft.com/en-us/um/redmond/projects/kinectsdk/

of detected joints is relatively low and only a rough position is calculated. Moreover the system has a large and noticeable delay (μ 218 ms) due to the complexity of the joint tracking computations.

2.4 Music Glove

Considering the recent developments in acquisition technology for remote hand movements tracking described in the previous sections, we have opted for a combination of several techniques. The following elements from previous research which are used in the interface are summarized:

- **Chung Lee's Wii controller technique.**
  Tracking up to 4 blobs of IR light.
- **Marrin and Paradiso their LED Baton**
  Creating a reliable IR source at the user's fingertips.
- **Sony's Move Diffuser**
  Creating an diffuse blob of light to transform a divergent IR-LED source to a omnidirectional marker.

In the next chapter we will further justify the choice for selecting these hardware elements for the use in our digital musical interface.

Chapter 3

Interface Development

Developing a hand gesture driven DMI involves hardware and software design. This chapter addresses the different steps involved in the design and presents comprehensive summary of the elements used for the DMI. During the development, an iterative process was adhered to for which evolutionary prototyping was used to incrementally improve the design.

3.1 Hardware Development

Because hardware often sets restrictions on software rather than vice versa, first the hardware for the interface is determined . In the next section requirements and constraints of the hardware is specified.

3.1.1 Requirements

The specification of requirements captures *what* the system is expected to provide: the user requirements. It states in plain language what is required for the end user. The user requirements are divided into functional requirements and constraints. The first describes the functional services of the system, the second the constraints which the system should satisfy.

Functional Requirements - Qualitative	**Functional Requirements - Quantitative**
Positional tracking of one or two hands	(x,y) per hand or finger

Constraints - Qualitative	**Constraints - Quantitative**
Fast tracking, i.e. without significant delay between movement and system processing	$<$ 50 ms
Accurate tracking, i.e. high resolution motion detection and high refresh rate	detectable change of 1 mm finger-movements at operation distance at 100 Hz
High degree of movement freedom, i.e. range and orientation: hands must be trackable with stretched arms and in any orientation	195 cm left-to-right and up-down span (95th percentile male radius of fingertip boundary [4]) at 180 degrees in horizontal and vertical planes
Affordable and easily available hardware	$<$ 20 euro
Portable	$< 1\ dm^3$
Easy to build	$<$ 1 hour build time

3.1.2 Architectural Design

Based on the requirements listed above, this section presents *how* the system should provide these services while satisfying the constraints by the describing of our re-iterative design of the interface. First the main hardware is determined after which 3 consecutively improved prototypes are described.

Wii Remote The main component of the hand tracking system consists of an optical sensor: the Nintendo Wii remote controller (Figure 3.1). It was originally designed to be a gaming controller and holds a set of sensors such as a gyroscope, accelerometer and an infrared camera. Internal hardware processes the output from the camera to detect and track blobs of infrared light. Up to 4 blobs can be tracked by the system simultaneously. The output of the tracked blobs are specified as:

$$\mathrm{O}_{(t)} = \{\ \mathrm{B1}_{(t)}, \mathrm{B2}_{(t)}, \mathrm{B3}_{(t)}, \mathrm{B4}_{(t)}\ \}$$
$$\text{where } \mathrm{B}_{i(t)} = \{\ \mathrm{x}_{(t)}, \mathrm{y}_{(t)}, \mathrm{s}_{(t)}\ \}$$

Hence of the ith LED at time t the horizontal and vertical position relative to the Wii controller and the blob size are produced. The blob size output is rescaled to range with 6 values, thus only a low resolution depth approximation is produced. Further, the device attempts to track each blob and assigns an unique position in the output for each blob. The output is transmitted via Bluetooth and can be received and processed by using a personal computer. [32].

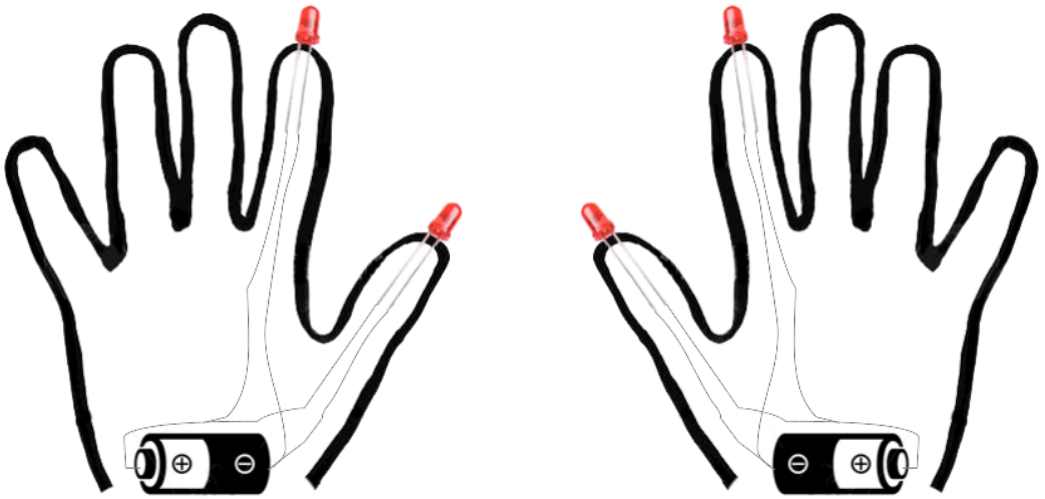
Johnny Lee [33] suggested a setup to utilize this Wii remote property. It functioned by adding reflective tape to fingertips which reflect light emitted from an array of IR-LEDs back to the Wii remote's camera. Four fingers could thereby be tracked. Although a

FIGURE 3.1: Top and front view of a Nintendo Wii remote controller.

fluent result without delay was realized, the angles of operation and range were limited in this approach (see Section 3.1.2). Further, the reflective tape does not reflect the infrared light in all directions equally due to bending of the reflective material, causing occasional loss of the signal.

IR-LED Source To overcome the shortcomings in Johnny Lee's approach, the idea of Marrin and Paradiso [38] was adapted to strenghten the input signal to the Wii remote by sending out infrared light directly from the fingertips. For this purpose we designed a glove with IR-LEDs and a power source attached. In Figure 3.2 the initial design of this glove is shown.

FIGURE 3.2: Initial design of the glove. A battery in the wrist powers IR-LEDs situated at the thumb and index finger. Two gloves emit a total of 4 IR sources.



First Prototype The first glove prototype used one 25 mW LD271 IR-LED[1] per finger. To verify that this configuration was sufficient to meet the requirements, two

[1] wavelengths: $\lambda > 760$nm

important quantities were assessed: 1) range[2] and 2) angles of operation[3].

Results indicated an improved range and angles of operation compared to the reflective method. A comparison study where the method of Johnny Lee was replicated, provided a maximum range of 0.75 meters and a 40 degree angle of operation. The first glove prototype improved this to 0.85 meters and a 60 degree angle of operation.

During testing of the first prototype, a disturbance in the output of the tracking system was noticed. When 2 or more infrared sources became too close to each other, they became indistinguishable which resulted in the switching of the output order of the two blobs. In Section 3.3.2 we propose a software solution to handle this problem of switching.

Not all requirements were met using this combination of hardware. Although the setup provided fast tracking (μ delay: 49.6 ms), accurate tracking (μ refresh rate: 98 Hz), both the maximum operation angle ($60°<180°$) and distance were not satisfactory. Stretched arms were not possible at a distance of 0.85 meters.

The requirement of accuracy is further not met by using the Wii remote controller. At a 195 cm stretched arm length window and with a resolution of the Wii remote's camera of 1024x768 pixels, this results in an accuracy of $\frac{1950mm}{1024px}$= 1.90 mm/px horizontally and $\frac{1950mm}{768px}$= 2.54 mm/px vertically. Hence the user must at least move 1.90 mm in the horizontal plane or 2.54 mm in the vertical plane in order for the system to detect a change in movement.

The hardware is affordable with a maximum cost of 17 €. For around 10 € a Wii remote controller can be obtained. The cost of the gloves is around 2 € whereas the LEDs and circuitry are valued at a maximum of 5 €. Furthermore the glove is easy to assemble within 1 hour.

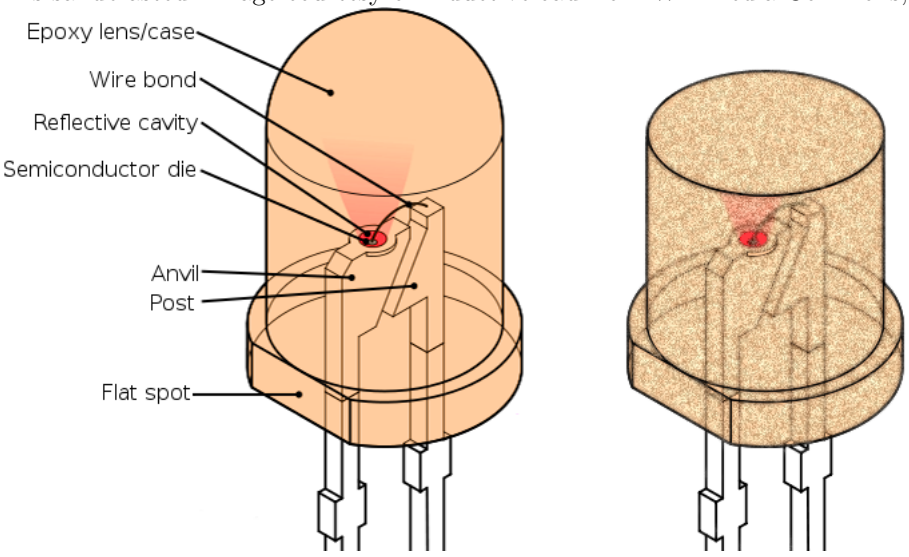
Second Prototype In order to increase the range and angle of operation, the IR-LEDs were modified. In Figure 3.3 an abstract representation before and after modification is presented. Modification was achieved by removing the epoxy lens. The regular LED emits a slightly divergent beam, whereas the modified version disperses the light multi-directionally. The more spread the beam of light the better it can be recognized

[2]Determining the maximum distance for proper operation was done by drawing circle via a visual feedback system while the user walked backwards. Whenever a circle failed, the current distance to the Wii controller was measured as the maximum distance of operation. For the maximum angle, the user was situated at the maximum distance, minus 10 cm. One finger of one glove was pointed towards the Wii controller. Next the hand was rotated, either to the left, right, upwards or downwards. When the signal disappeared, the angle was measured relative to the starting point.

[3]Measured relative to pointing directly at the Wii controller at 0 degrees. Directions measured were left, right, up and down. For all directions the maximum angles were equal.

from more directions, improving the angle of operation. Furthermore, the surface was sandblasted [4] in order to further increase the dispersion of light.

FIGURE 3.3: Abstract representations of LEDs. On the left a typical Light Emitting Diode (LED). On the right the modified version where the lens is removed and the body is sandblasted. Image courtesy of Inductiveload from Wikimedia Commons, modified.



The effect of this modification can be shown in the visual light spectrum by using Red-LEDs[5] (see Figure 3.4). This comparison indicates that the maximum angle of operation will be increased because the light is dispersed at a greater angle. Moreover, the intensity of the beam does not seem to be largely effected by the dispersion, hence the maximum distance of operation is expected not to reduce.

Verification of these modifications confirmed these predications. The range remained equal (0.85 m) but the operation angle was increased to a 100 degree view in all planes.

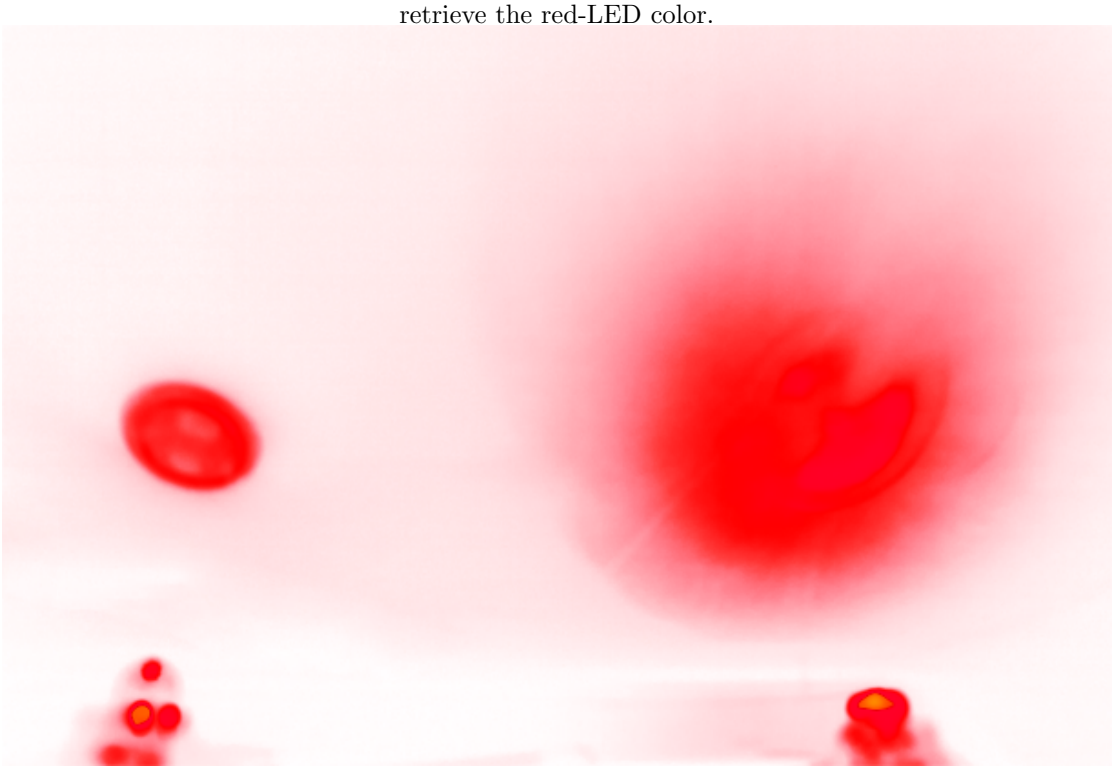
However, both the range and angles of operation were not satisfactory. To increase the range, the amount of electrical current through the LEDs was increased to 125 mW. This effectively increased the maximum range to 3.00 meters. At this distance the user can use a stretched arm, covering a span of 195 cm from left-right and up-down.

Third Prototype In order to increase the angle of operation, the idea of diffusion used in Sony's Move controller was applied where a semi-translucent plastic sphere diffuses the LED's light underneath, enabling the sensor camera to track the device irregardless of the angle of operation.

[4] Given our constraint of easiness to build, using regular sandpaper instead of sandblasting equipment also suffices. A rough surface should be the result.

[5] wavelengths: $610\text{nm} < \lambda < 760\text{nm}$

FIGURE 3.4: Photograph of two light beams (top 2 blobs) from a regular Red-LED (left) and a modified Red-LED (right). LEDs (bottom 2 blobs) were placed in a dark room, 5 cm in front of a white papered wall. The photographic was taken with a Canon EOS 400D at ISO 100, F 5.6 and 1/4 exposure time. The image's colors were inverted to change the black room background to white. Further, hue was inverted next to retrieve the red-LED color.

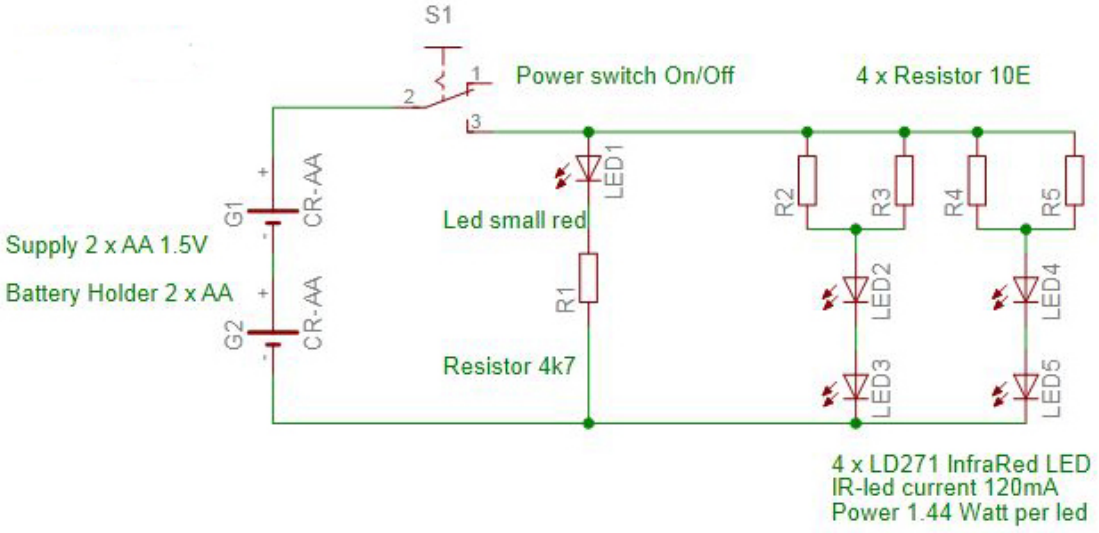


In Figure 3.5 a diffuser, designed especially for our interface, is shown. It is made from a semi-translucent plastic, and contains small particles which disperse the light to a all angles. Because the light is partially absorbed by the plastic, two LEDs per diffuser were required to compensate for the loss of light.

FIGURE 3.5: A diffuser: a piece of diluted plastic with two holes for inserting standard 5mm LEDs. The divergent LED light is emitted through the plastic omnidirectional, creating a 'blob' of light rather than a beam.

Testing this third prototype delivered satisfactory results, meeting the requirements for range and angels of operation. The range was slightly reduced to 2.30 meters, however the angle of operation was increased to a view of 180 degrees in all planes. Both proved to be sufficient, allowing gestures with stretched arms, pointing in every direction. In Figure 3.6 the circuit diagram of the electronics in the third prototype is shown.

FIGURE 3.6: Circuit diagram of the final glove.



3.2 Software Development

The behavior of a DMI is determined by software. In the context of DMIs, three main functions of the software can be recognized. 1) Continuous data acquisition of human generated control signals and 2) transforming hardware signals to auditory information and 3) output it as audio signals. As will be further detailed below, we have considered two options for the development of the required software: 1) develop from scratch and 2) integrate existing modules. A detailed specification of the software requirements is given in the next section.

3.2.1 Requirement Specification

Functional Requirements - Qualitative	**Functional Requirements - Specifics**
Interface with relevant hardware	Bluetooth Wii controller data
Transform and output control signals to auditory information	MIDI or audio signal
Transform control signals to visual information	Visual feedback of (x,y) per IR blob

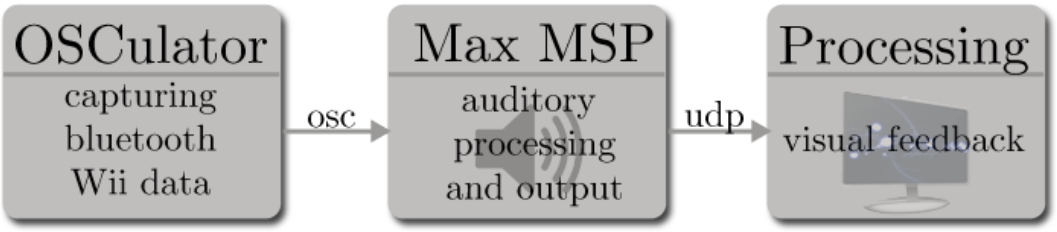
Constraints - Qualitative	**Constraints - Specifics**
Versatile	Adjustable for adding functionality
Efficient	No computational delays
Distributable	Compiled & Open Source
Modular	Easily replaceable individual components
Platform independent	Windows and Mac OSX
Gesture recognition	$>$ 99 % recognition rate

3.2.2 Architectural Design

No single software framework was found which met all requirements, apart from developing a framework from scratch. Multiple different frameworks were found which each partially provided the required functionality. This approach to combine these frameworks was chosen above building a new framework from the ground up in order to be able to quickly build prototypes.

The requirements are split to 3 software frameworks. In Figure 3.7 an overview of the chosen frameworks and their interactions are visualized.

FIGURE 3.7: The architecture consisting out of the three software frameworks, their respective functions and data flows.



Data Acquisition OSCulator is a program that is able to capture Bluetooth data from the Wiimote and sends it to a wide variety of different programs via different protocols [52]. The program is not freeware nor platform independent, however multiple freeware solutions for both platforms exist [30].

Open Sound Control was chosen as a platform independent protocol to transfer the Wiimote data over the intra/internet. Designed as an alternative to MIDI at CNMAT, it features higher resolution data transfers which are distributed faster compared to MIDI.

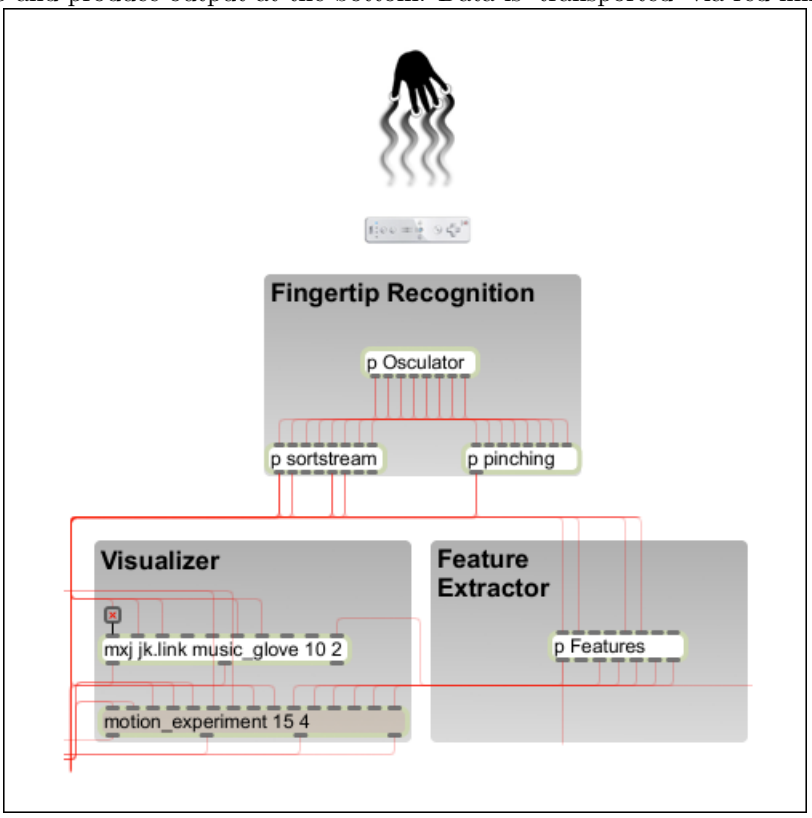
Data Processing Max Msp is a modular visual programming language for multimedia designed at IRCAM by Miller Puckette [3]. Given the OSC input from OSCulator, it can perform operations on the data and transform it to audio information. The framework is multi platform. Further, a build in Java editor and compiler provide the the capability to implement novel algorithms.

Inside Max Msp, objects are represented visually. In Figure 3.8 a part of the final implementation is shown. This can either be compiled as a stand-alone application, or released editable as source code. Components can be easily replaced, as long as the inputs and outputs remain equal.

Note that Max MSP is not freeware. Source code cannot be edited without a purchase, however compiled software can be distributed freely.

Data Visualization Processing is a Java-based programing framework specifically designed for efficient visualizations [18]. It can receive this information by UDP messages sent over the intranet by Max MSP. Although Processing could also receive motion information directly from OSCulator, the route via Max MSP was chosen such that also auditory information could be sent in synchrony when required. The functionality implemented by the software program provides the user with visual feedback of the hand position in space.

FIGURE 3.8: A part of the implemented structure in Max MSP. Three areas are shown in grey, each with a unique function. Objects within these areas receive inputs at the top and produce output at the bottom. Data is 'transported' via red links.



3.3 Final System

3.3.1 Hardware Result

In Figure 3.9 the final result can be observed. A glove with two infrared sources attached to the index finger and thumb are powered by a hidden power source inside the wrist. The hardware approaches the requirements and constraints. The Wii remote tracks the hand quickly (delay: < 50 ms, refresh rate: 98Hz), which according to a user study did not add a disturbing experience of lag or was not even noticeable at all. The resolution of motion detection was not met (2.54 mm/px $>$ 1.90 mm/px $>$ 1.0 mm/px), however according to user studies the maximum resolution of 2.54 mm/px was sufficient enough for proper movement generation (4.03/5 points, stdev 0.99 points). Furthermore, the glove is comfortable to wear and its presence feels to disappear when it's used (see Section 5.4.2). At a distance of 2.30 meters, users can have stretched arms (195cm in the horizontally and vertically) in every direction (180 degrees) while the Wii remote can still track the hand motions. Further, the hardware is affordable (< 20 €), easy to build (< 1 hour) and portable ($0.2\ dm^3 < 1\ dm^3$)

FIGURE 3.9: The final version of the glove (top). The power source (bottom) is hidden inside the wrist.

3.3.2 Software Result

On the software side a framework was created consisting of 3 separate components. The framework meets the requirements. It captures gestural control signals of the user via Bluetooth, which is transformed to an audio representation within Max MSP (see Chapter 6). The gestural data is further visualized via Processing, providing visual feedback to the user (see Section 3.3.2).

Almost all constraints are met. The framework is versatile in the sense it is not restricted to a fixed set of functions. Novel algorithms can be implemented if required.

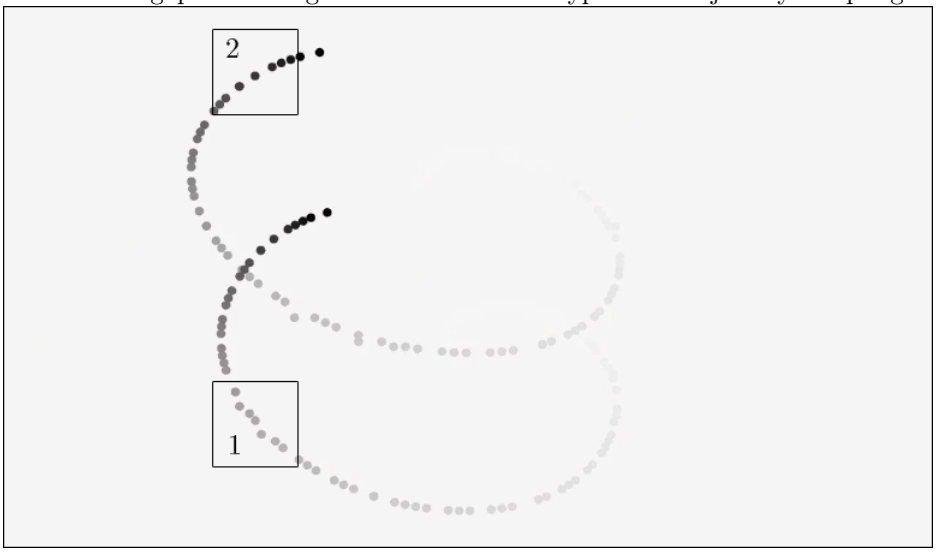
Furthermore, the modular approach enables developers to replace or add components to improve or alter the functionality. The software runs efficiently without noticeable delay (see Section 5.4.2). Besides being platform independent, the software can be distributed as a compiled package, including the editable source code.

Hand gestures are tracked and they can further be recognized. Specific symbols can be drawn and classified with a satisfactory performance (>99%) (see Chapter 4). Furthermore, motional features can be extracted in real-time which can be used as holistic continuous gestures for sound control (see Chapter 5 and 6).

Visual Feedback The visual feedback provided to the user shows positional data faded over time (see Figure 3.10). Minor issues exist with the accuracy and signal gaps regularly occur. This might be caused due to small finger tremors or noise in the device. A possible solution is to filter the output by for example a smoothing filter [36].

Pinching An important final design feature eradicates the undesired property of the blob switching output from the Wii controller's tracking algorithm. Instead of allowing the sources to switch, they are forced as a interaction design to a single source when they come too close to each other. Because the user has infrared sources on the index finger and thumb, this enables the user to 'pinch'. This is a similar to pen up/down interactions in tablets [17] which provides another level of interaction for the user.

FIGURE 3.10: The visual feedback software end result: the x and y position of 2 IR blobs from one glove are tracked and rendered as individual black dots at 50Hz. Over a timespan of 1,5 seconds, each dot fades from black to white. Two areas of interest are highlighted. Number 1 shows a slight inaccuracy in the tracking. Number 2 shows occasional gaps in the signal. Both issues are typical in trajectory sampling.



Chapter 4

Analytic Control Gestures

Abstract

This chapter reports on the development and analysis of a set of analytic control gestures for the DMI, resulting in a repertoire of 17 symbols. As a proof of concept, they are distinguished by a Hidden Markov Model, trainable by the user with only a few samples. Furthermore, the performance is compared to results from a k-nearest neighbor classifier, specialized in recognizing similar gestures. The results are promising: a recognition performance of 100% can be achieved after a few hours of practice.

Analytic control gestures are motions intended for controlling discrete events in a digital musical instrument. They are like buttons or switches, which trigger either a music or system event (see Chapter 6). The gestures are much alike symbols which can be drawn in the air. In the following section the requirements for these gestures is specified. After this the repertoire design is described, providing the design rationale behind the choice of symbols. Next the classifier software is determined whereafter it is evaluated on its performance given the gesture repertoire. The results of this exploratory study is then compared to a second classifier in order to determine if the chosen classifier for our system is performing properly.

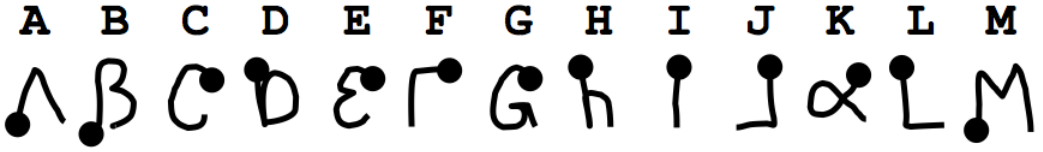
4.1 Requirement Specification

- Functional Requirements
  - A suitable repertoire (10+) of 2D hand gestures.
  - A classifier system which recognizes these gestures.
- Repertoire Constraints
  - Gestures should feel natural/intuitive, i.e. fluently drawable.
  - Gestures should be easy to learn and reproduced by the user.
  - Gestures should be visually pleasant. During live musical performance, esthetics are important.
  - Gestures should be easy to distinguish.
- Classifier Constraints
  - A high recognition rate of the gestures, i.e. above 99%
  - Fast recognition, i.e. no delay between drawing and classification.
  - Pre-defined and Java-based, or as a plugin for Max MSP.
  - The classifier should be trainable by the user, i.e. a low amount of required training samples.

4.2 Gesture Repertoire Design

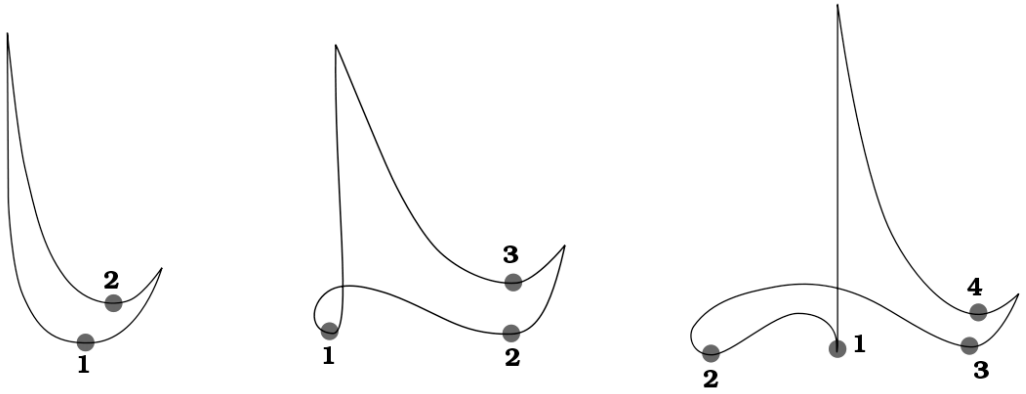
In previous research, easy-to-use and distinguishable 2D gesture symbol sets are already developed. The most prominent set contains the unistroke gestures from Goldberg[21, 22], depicted in Figure 4.1. They are characterized by their creation as a single stroke. In other sets using multiple strokes, uncertainty is introduced, making it harder to be distinguished by the system. The major advantage of unistrokes is that it eliminates this uncertainty, also known as the segmentation problem. After its success, many other symbol sets were based on Goldberg's idea [27, 55].

FIGURE 4.1: A subset of Goldberg's and Richardsons's unistroke: a simplified alphabet gesture repertoire.



With our interface, it is possible to draw such unistrokes by utilizing the pinching property (see Section 3.3.2) as a pen up or down event. When the user starts or releases a pinch, it respectively indicates the beginning or the ending of a gesture. However Goldberg's set of unistroke gestures do not adhere to our requirements. A more musically intuitive and visually pleasant set is required. Therefore the unistroke idea is combined with the musical gestures made by conductors, such as depicted in Figure 4.2. This type of gestures are positively associated with musical performances and hence it is assumed that this would be a good starting point for our repertoire creation.

FIGURE 4.2: Example 2D traces of conductor's gestures. Starting at the top, a downward movement is made whereafter a directional hopping movement is continued upward. Numbers indicate points in time where a beat occurs [37].



The result is an initial repertoire of 12 gestures shown in Figure 4.3. Gestures are created by pinching at the top whereafter a downward movement precedes an upward directionally distinct motion. As will be discussed in Section 4.4.5, the initial repertoire is modified to improve the recognizability by the classifier. The modified set is shown in Figure 4.4.

FIGURE 4.3: The initial repertoire of analytic control gestures.

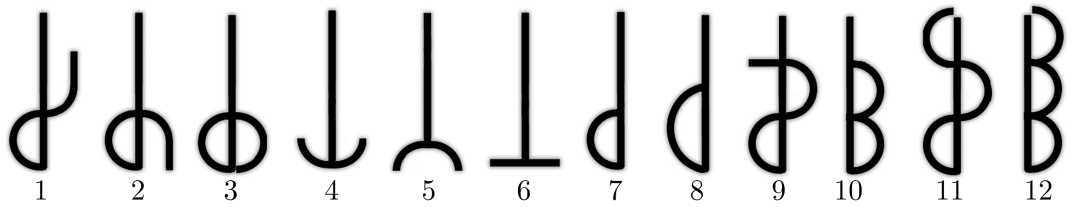
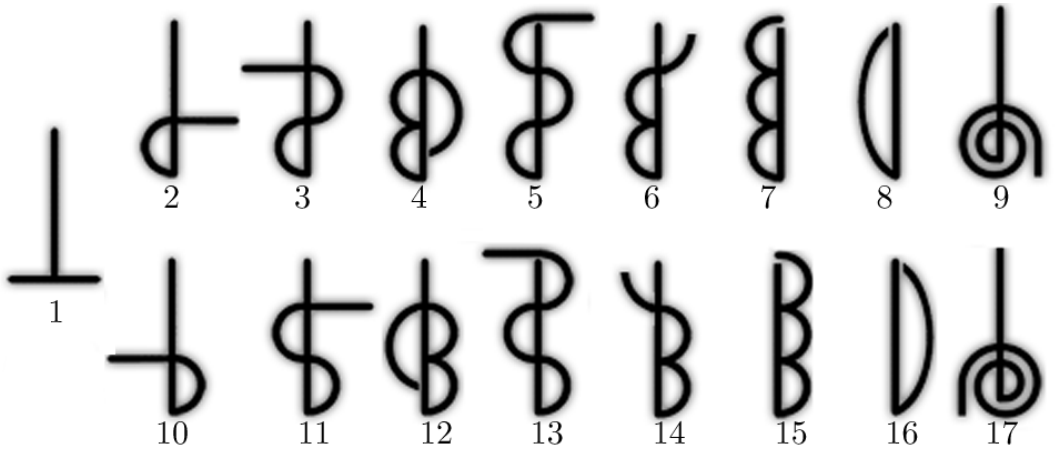


FIGURE 4.4: The final repertoire of analytic control gestures.



4.3 Classifier Determination

In order to classify the analytical control gestures, IRCAMs predefined Hidden Markov Model (HMM) based classifier is used [8]. The classifier is specifically designed for artistic performances and features incremental[1] gesture recognition especially designed to be trained with a *single* example. It is very well suited for consistently performed temporally differing gestures, in which it is known musicians are proficient [46, 47].

4.3.1 Classifier Operation

The classifier 'follows' a gesture by calculating at each subsequent point in time an updated likelihood value for each class. In this section the algorithmic workings thereof are described.

Learning First the model has to be trained by providing a single example per class. Assumed is that the gestures can be represented as a multidimensional temporal curve[2]. The learning procedure for a single class is summarized in Figure 4.5.

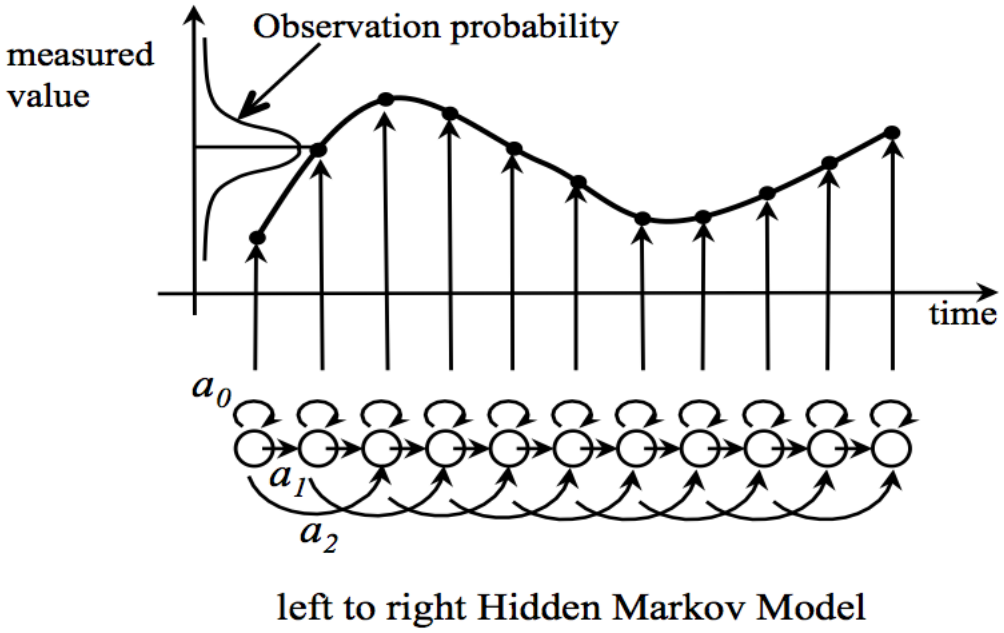
Each state i outputs an observable O with a probability b_i which follows a normal distribution in the following manner:

$$b_i(O) = \frac{1}{\sigma_i\sqrt{2\pi}} exp[\ -(\frac{O-T_i}{2\sigma_i})^2\ ]\ ,$$

[1] At each point in time, the classifier outputs a likelihood distribution of target classes. This distribution is updated after new evidence is presented to the classifier. Hence gestures are incrementally recognized over time.

[2] For example the values of $x_{(t)}$ and $y_{(t)}$ from the output of Wii remote controller are temporal curves.

FIGURE 4.5: The learning procedure: modeling a training sample in a left-to-right HMM. Figure taken from [8].



where T_i is the value of a temporal curve at time point i in the training sample and σ_i is the standard deviation of T_i between training samples. Since σ_i does not exist when only one training sample is present, it is estimated using prior knowledge of the context. This knowledge can be obtained in for example a user study where the average standard deviation of an obtained gesture set can be calculated to serve as σ_i.

Furthermore, transition probabilities between states are restricted to a_0, a_1 and a_2 as depicted in Figure 4.5. Satisfying the constraint that $\sum_{i=0}^{2} a_i = 1$. In most applications the following transition values suffice:

$$a_0 = a_1 = a_2 = \tfrac{1}{3} \qquad \text{or} \qquad a_0 = a_1 = 0.25 \text{ and } a_2 = 0.5.$$

Decoding The decoding follows a standard forward procedure HMMs. Let $O_1, O_2 ... O_t$ be the observation sequence of a gesture. In order to derive the probability distribution at point t in time, $\alpha_t(i)$ is computed by initialising:

$$\alpha_1(i) = \pi_i b_i(O_1) \qquad 1 \leq i \leq N \ ,$$

where π is the initial state distribution, and b is the distribution of observation probabilities. Hereafter $\alpha_i(t)$ is inducted by:

$$\alpha_{t+1}(i) = [\ \sum_{i=1}^{N} \alpha_t(i)a_{ij}\ ]\ b_i(O_t) \qquad 1\leq\ t\ \leq\ T\text{-}1\ ,\ 1\leq\ j\ \leq\ N\ ,$$

where a_{ij} is the state transition probability distribution. When $\alpha_i(t)$ is computed, the likelihood of the observation sequence and the time progression in the test sample can be calculated by:

$$\textit{time progression index(t)} = \textit{argmax}[\alpha_i(t)], \text{ and}$$

$$\textit{likelihood(t)} = \sum_{i=1}^{N} \alpha_i(t).$$

4.4 Exploratory Study

To determine the recognizability of the gesture repertoires given the chosen HMM classifier, exploratory studies were performed. For this purpose, data was collected and classified in a simulation based on repeated random sub-sampling validation.

Data was collected in 4 stages. After each stage, either the gestures, the glove or the visual feedback was modified to check improvements in performance. The following conditions were used for each data collection:

1: Glove prototype 2, initial gesture set, with tracks, 50 samples/class.

2: Glove prototype 2, final gesture set, with tracks, 20 samples/class.

3: Glove prototype 2, final gesture set, without tracks, 20 samples/class.

4: Final glove, final gesture set, without tracks, 20 samples/class.

For an explanation of the use of tracks, see the next section.

4.4.1 Method

Participants Up to 3 subjects participated in each data collection. All subjects were right handed and had prior knowledge of the system's inner workings and all were familiar with the interface and goals of the study.

Materials The 2nd and 3rd generation glove prototypes were used. Both consisted of 2 sources of IR light, placed on the index finger and thumb on a right handed glove.

No fixed setup was used, however identical interaction situations were realized for each session. A solid stand with a variable height was positioned in the room. On this stand

an Apple Cinema HD Display (23-inch LCD @ 1920 x 1200 pixels) was placed, connected to a Macbook Pro 13" 2010 model (2,4 GHz Intel Core Duo, 4GB 1067 MHz working memory). The Wii remote controller was placed on a leveled surface at the same height of the center of the screen, pointing towards the participant.

A modified version of the visual feedback was presented by Processing. The background color of the visualization was black. Positions of the participants' two fingertips were displayed as red circles (ø10 pixels). When the participant pinched, both circles were displayed a singular white circle of equal size. Further, participants were able to see a fading trail of these circles representing their previous movement positions. These positions were faded entirely after 1.5 seconds. Positional data per finger and pinch information ($x_{(t)}$, $y_{(t)}$ and $p_{(t)}$) was saved to disk at 50Hz.

Procedure Participants were situated equidistant across data sessions at 1.5 meters in front of the screen. Target gesture classes were presented in a randomized order. A red dot (ø50 pixels) appeared at a random position on the screen[3]. Participants were instructed to move over this dot before moving at the starting position of the gesture. At this position, they were required to pinch and complete the gesture sample. The sample was ended by a pinch release.

On average, a single recording took 1.5 hours to complete for gesture set 1 and 0.75 hours for gesture set 2. To counter fatigue, subjects could pause the recording when they required a brake. When a sample drawing failed due to user errors, the participant could redo the gesture by pressing spacebar.

In the conditions with tracks, guiding boundaries of the target gesture were shown, as depicted in Figure 4.6. The other conditions presented a small bar at the bottom, highlighting the target gesture.

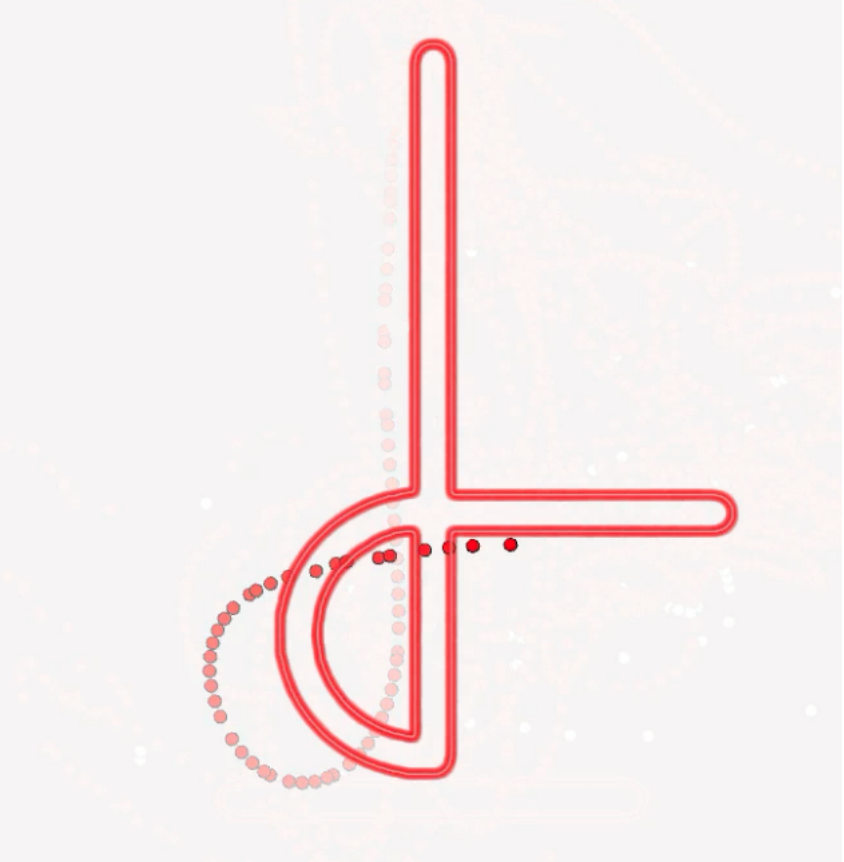
4.4.2 Segmentation & Resampling

To accommodate the data for simulation usage, it was segmented and resampled. Segmentation is desired to solely obtain the data points ($x_{(t)}$,$y_{(t)}$) which constitute the relevant part of the gesture. Resampling is necessary for proper prototype creation from multiple samples.

Segmentation is achieved by exclusively selecting the parts in the data where the users pinch. Furthermore, when multiple pinches exist in the data, the whole sample is discarded because the HMM can only cope with unistroke gestures.

[3]This was implemented to avoid a bias in starting positions.

FIGURE 4.6: A screenshot of the condition with tracks while a gesture is made. For this image, the background color is inverted.



Resampling is performed in two ways: spatially and temporally. The former discards velocity information and reorders gestural coordinates equidistantly throughout the gestural form. The latter preserves the velocity information and results in gestures with non-equidistant coordinates.

4.4.3 Simulation

A simulator was programmed in Java and loaded inside Max MSP. It derives a prototype training sample per class, trains the classifier and writes the results to file.

Prototype Generation Because the HMM classifier is trained by a single example per class, a perfect prototype is derived from multiple samples in order to optimize recognition results. The calculation is performed as follows:

$$x^P_{(t)} = \frac{\sum_{i=0}^{i} x^S_{(t)}}{i-1} \qquad \text{and} \qquad y^P_{(t)} = \frac{\sum_{i=0}^{i} y^S_{(t)}}{i-1},$$

where $\{x^P_{(t)}\ ,\ y^P_{(t)}\}$ and $\{x^S_{(t)}\ ,\ y^S_{(t)}\}$ are the coordinates in the prototype and samples respectively.

Training For each class, the classifier is trained with a prototype derived from random samples not used in the test set. Next the classifier is tested by randomly providing a test sample per class. When all classes are tested, the procedure is repeated for a total of 150 times.

Exploratory simulations were run in order to approximate optimal parameter settings. The range of {1,2,5,10,15,20} samples per prototype was evaluated. For {spatial, temporal} resampling the range of {5,10,15,20,25,30,35,36,37,38,50,75} as the number of data points was evaluated. Results indicated that at a 15 sample prototype on temporally resampled data to 37 data points proved the best classifier performance. Therefore all simulations are performed using these settings.

4.4.4 Results

Condition 1 Three subjects {S1, S2, S3} participated in this condition. The performances in the simulation were 85.38%, 84,71% and 64.45% respectively.

Condition 2 One subject {S1} participated in this condition. The performance in the simulation was 99.15%.

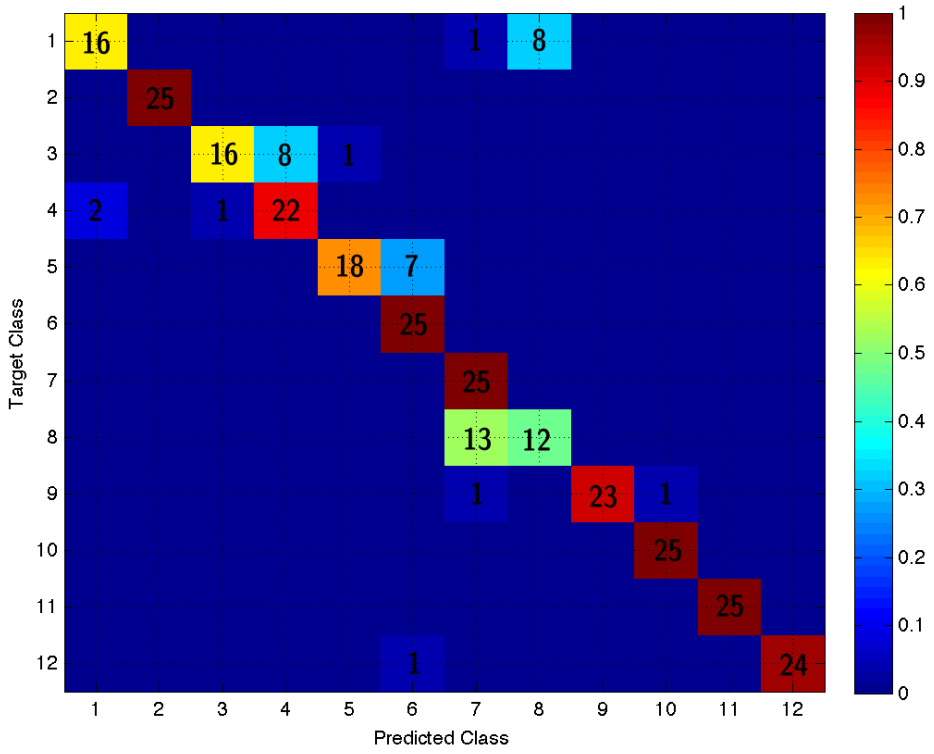
Condition 3 Two subjects {S1, S2} participated in this condition. The performances in the simulation were 98.84% and 94.87% respectively.

Condition 4 Two subjects {S1, S3} participated in this condition. The performances in the simulation were 100% and 80.56% respectively. A follow-up simulation with a *single* sample prototype from Subject S1, resulted in a performance of 99.15%.

4.4.5 Discussion

The performances in condition 1 were not satisfactory since the recognition performance is required to exceed 99%. Therefore ways improvement were sought by investigating the confusion of the classifier between classes. In Figure 4.7 an example confusion matrix is depicted used for this analysis.

FIGURE 4.7: Confusion matrix from results of subject S1 in condition 1, over 25 train/test cycles. Each cell *m(i,j)* contains a value which equals the number of times the classifier labels a target class i as a predicted class j . With perfect classifier performance, the values in cells *m(i,i)* should be equal to the number or train/test cycles, whereas in all other cells the value equals 0. Values in each row should always add up to the number of train/test cycles. Colors indicate relative performance $\frac{m(i,j)}{numberoftrain/testcycles}$.



The confusion matrix summarizes the classification results per target class. Hence it provides insight how the classifier confuses classes with one another. According to Figure 4.7, classes {2, 6, 7, 10, 11} are perfectly recognized. Other classes are not properly distinguished by the classifier.

Multiple causes for this confusion may exist. First some gestures might not have been easy to perform, resulting in erroneous samples and high sample variance. Further, some gestures might not have been optimally distinguishable by the system, for example the gesture classes in Figure 4.8.

FIGURE 4.8: All samples from subject S2 in condition 1 of class 7 (left) and class 8 (right) drawn on top of each other.

To improve the recognizability, the gesture repertoire was adjusted by modifying the classes {1, 2, 7, 8, 9, 10, 11} by adding more directional differentiation. Other classes {3, 4, 5} were removed. Furthermore all classes[4] were mirrored to increase the total set size to 17. Since the classifier is sensitive to directional differences, it is expected not to effect the performance.

This modification of the repertoire led to condition 2. The performance of the simulation on the corresponding data set satisfies the requirements. However, a supporting gestural track is still present in the visual feedback. Hence it was removed to investigate the participant's ability to perform gestures without support.

This led to the results from condition 3. Though performance for subject S1 dropped with an absolute percentage of 0.31%, this indicates that the gestures can be drawn without guidance while maintaining the performance level.

The final condition differed from the previous condition in the use of a different glove. The increased angle of operation of the interface makes it less restrictive to perform gestures. This increases the precision of the motions by allowing the most preferred posture of the user. This in combination with learning effects, optimizes the performance of subject S1.

[4]Except for class 1 to avoid directional ambiguity.

4.5 Classifier Performance Comparison

To assess the performance of the HMM classifier compared to other classifiers, a comparison with a baseline gesture recognition classifier was performed. This type of classifier employs the well-known k-nearest neighbor technique (knn). By comparing the performance of this classifier to the performance of the HMM classifier, it can be established whether both classifiers can achieve similar recognition rates and robustness. The knn classifier uses a number of prototypical samples p_i with known classification c_i to classify a new, unknown, test sample x. Classification is performed by computing the match between each p_i and x and subsequently using majority voting on the k best matching prototypes [15].

The match $m(p, x)$ is computed using the Euclidean distance between the feature vector representations of both p and x. The feature extraction technique has been extensively researched in our department for the recognition of various types of pen-input data, such as handwriting and sketching [59]. Each gesture trajectory is spatially normalized and resampled to 30 (x, y) coordinates. The feature vector is extended with the running angle $(cos(\phi), sin(\phi))$ per coordinate pair and the angular difference $(\delta cos(\phi), \delta sin(\phi))$ per pair of running angles. The number of 30 coordinates is based on empirical evidence that a fairly complex Western character contains 5 velocity-based strokes and that 6 coordinates per stroke suffice for proper reconstruction. Note that this approach is distinct from the previously described technique of incremental recognition of the HMM classifier, since a complete gesture trajectory is required before processing of the gesture can be engaged.

Method For two subjects, S1 and S3, the classification performance of the knn classifier was determined using different k and different number of training prototypes *Np* per class. For each condition *(k,Np)*, 100 random prototype sets were selected. For each configuration, classification was performed on the remaining samples.

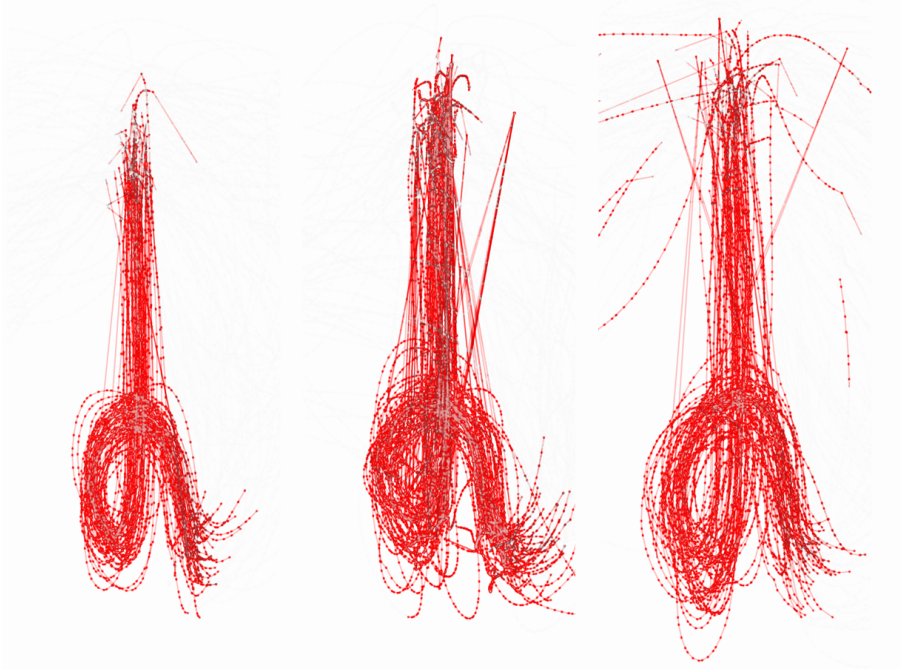
Results For k=1 the results were optimal. In the following table the results are summarized.

Discussion & Conclusion For the data of subject S1, both the KNN and HMM perform equally well. For the data of subject S3 however, the KNN outperforms the HMM. This indicates that the performance in the HMM is suboptimal for some datasets. The low performance could be caused by IRCAM classifier's inability to cope with data with a larger variances [8]. Visual inspection (see Figure 4.9) of the data confirms the

	Np	min	max	sd	avg
S1	1	94.74	100.00	1.21	97.88
	2	96.08	100.00	0.63	99.26
	3	97.58	100.00	0.52	99.59
	4	98.16	100.00	0.37	99.79
	5	98.82	100.00	0.26	99.88
	10	99.41	100.00	0.06	99.99
	15	100.00	100.00	0.00	100.00
	19	100.00	100.00	0.00	100.00
S3	1	75.20	94.49	4.14	88.40
	2	88.61	97.89	1.92	94.49
	3	91.82	98.64	1.42	95.81
	4	91.13	98.03	1.33	96.50
	5	92.47	98.92	1.22	97.10
	10	95.05	100.00	1.15	97.57

larger variance for subject S3. A possible solution to increase the performance of the HMM is to estimate the the σ_i parameter from the data per subject. However further efforts are needed to validate the cause and solutions and moreover to quantify inter-subject variance.

FIGURE 4.9: All samples from class 2 from subject S1 (left), S2 (center) and S3 (right) in condition 1. Samples are drawn on top of each other.



4.6 Conclusion

An analytical control gesture repertoire was created for the control of musical events. The goal was to create an easy to learn and distinguish natural/intuitive and visually pleasant set of symbols. Based on the proven concept of unistrokes [21, 22, 27, 55] and in combination with musically associated conductor's gestures [28].

During simulations the HMM based classifier proved proficient in distinguishing the gestures in the repertoire. For users making consistent gestures, which are commonly produced by musicians [46, 47], only a single training example suffices for a recognition performance of 99.15%. For less consistent users, practice could make perfect. The incremental nature of the classifier holds various opportunities. For example, the visual feedback can at each point in time indicate the belief distribution of the classifier. This could enable the user to release the pinch when the target class is recognized, hence speeding up the interaction or providing the ability to time the pinch release in with the music. A useful property of the classifier is that the classifier is trainable by novice users within reasonable time due to the low amount of samples (≥ 1)per class needed to derive a prototype. The supporting track in the visual feedback can help users to train themselves and the classifier accordingly.

Still the HMM classifier does not perform equally well for all users, hence further efforts are needed to improve the performance. The comparison study with a KNN classifier suggests that the same acceptable performance ($> 99\%$) can be reached for all users.

Chapter 5

A study on listener associations between musical changes and hand motions.

Abstract

This chapter reports on an experiment that was conducted to measure the effect of changes in dynamics, pitch, brightness, articulation, syncopation or rhythm on hand motions. This was measured by derivatives of motional features. Results indicated that many features of the motions are significantly affected by many musical parameters. Furthermore, between and within participants there is a low amount of motional type variance. These results will provide essential knowledge to create an intuitive musical mapping from hand motions to sounds. Moreover, supporting the musical embodied cognition thesis, it suggests that people have an internalized abstract representation of sound generating movements: a culturally shared representation of abstract sound features directly linked to movement.

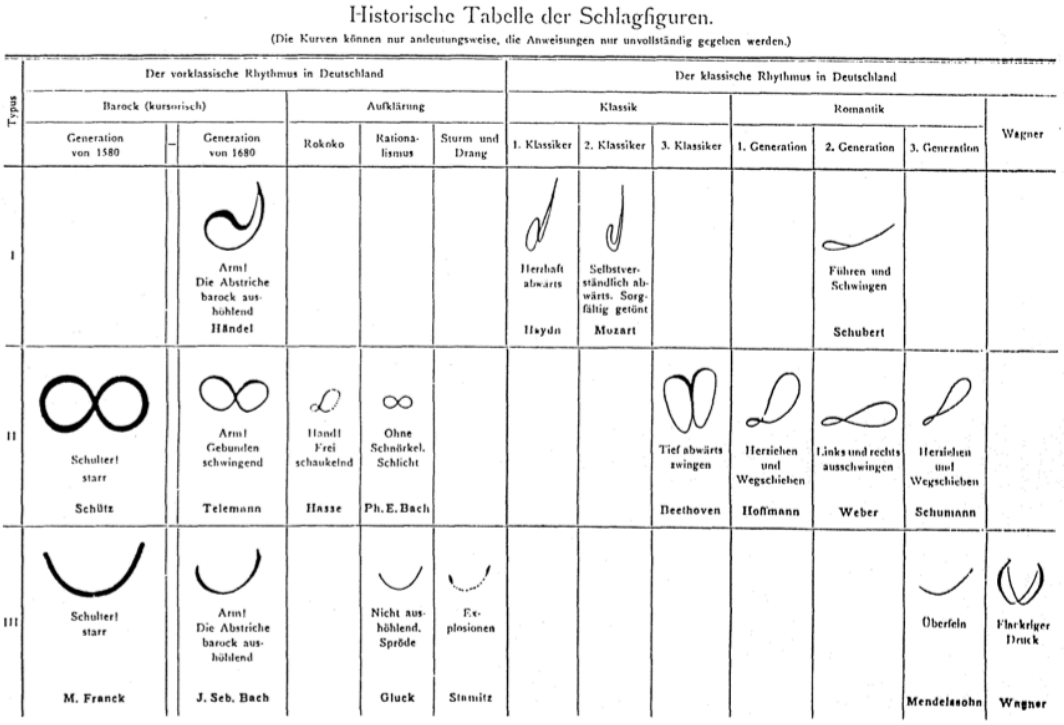
5.1 Introduction

Music and motion are interconnected: wherever music is present, motions are nearby. Literally by movement in the air, but also by people which tend to move to music [5, 11, 56]. Research shows that listening to music often is associated with body movements which are often synchronized with its periodic structure [51]. However to what extend and in which form both phenomena are related is highly debated [35]. One cause of this relation is thought to lie in the empirical world. When people generate music

through instruments, they perform a specific pattern of motion to cause changes in sound. Acoustic dimensions, such as pitch or loudness, are the result of a particular movement. These co-occurrences are thought to produce expectations inside humans. Associations might arise when either of the two modalities is activated [28]. Moreover, the notion of embodied music cognition assumes that music perception is based on a multi-modal encoding of auditory information that contains the coupling of perception and bodily action. This is opposed to a disembodied view in which only the perception-based analysis of musical structure gives musical meaning [35].

Music conductors represent a specific case in which hand movements are associated with music. Research to this relation dates back to 1928 when Becking made a classification of conductors's hand movements, while performing to different types of classical music [6]. In Figure 5.1 a categorization of conductor's gestures are summarized. Results indicated that, given a type of music to be conducted, a corresponding gesture was made.

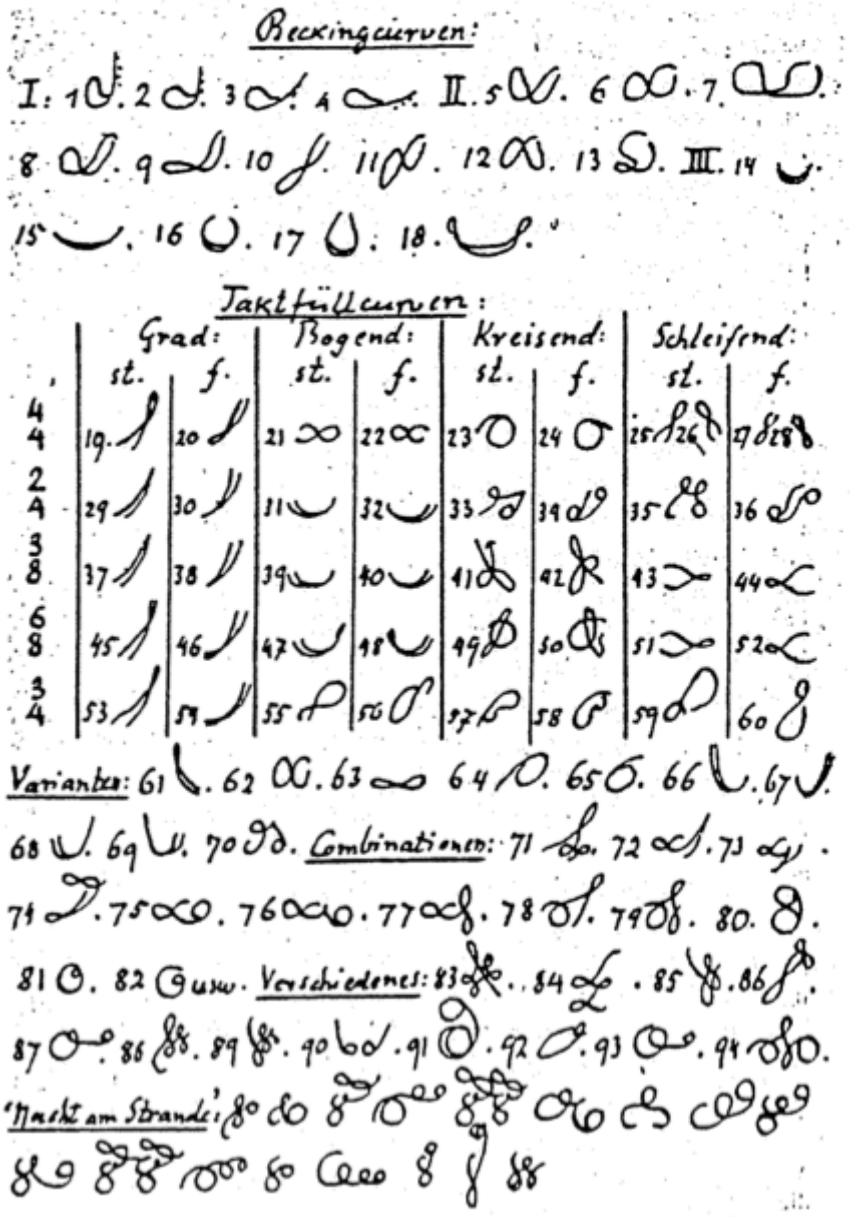
FIGURE 5.1: Becking's table of categorized conducting curves.



Similar to Becking, Sievers made a more extended categorization of movement curves associated with music. In Figure 5.2 his categorization findings are presented.

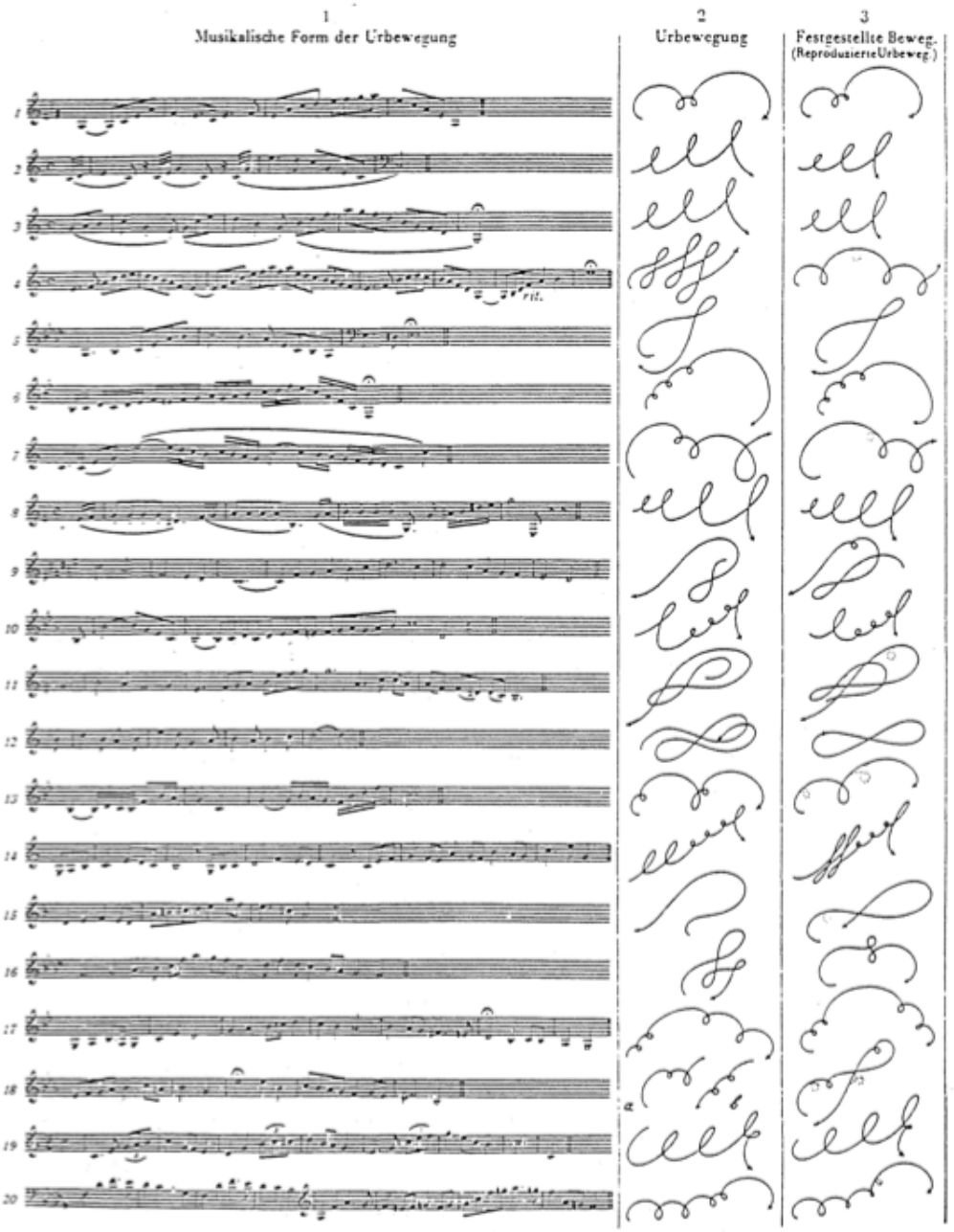
The methods Sievers used to obtain these curve categorizations lacked scientific rigor, but the underlying idea formed the starting point for Truslit [54]. In 1938 he published *"Gestaltung und Bewegung in der Musik"* in which an experiment is described that tests the hypothesis that motions will always co-occur with certain sound patterns.

FIGURE 5.2: Categorization of movement curves by Eduard Sievers.



Two subjects (N. and T.) participated. Subject N. chose a motion curve and sketched it on paper and carried it out with hand and arm movements. From this movement and written motion curve, subject N. then created an accompanying musical pattern. These musical patterns were then presented to subject T. who tried to determine the corresponding motion curves. The results are depicted in Figure 5.3 and indicated that 13 of these the recovered motions were almost identical. 6 Of them had only minor differences. These results were in favor for Truslit's hypothesis. Although an improvement on Sievers' method, the experiment also lacked scientific rigor. Methods were not specified and only 2 subjects participated [48].

FIGURE 5.3: Experiment results on the recovery of original motion from notated music.



Study by Eitan and Granot A recent study investigated how listeners associated changes in sound with perceived images of motion [16]. Subjects were presented with musical stimuli in which each stimulus had one musical parameter intensified or reduced. The subjects were asked to associate these melodic stimuli with imagined motions of a fictional character and to report several attributes of these movements on a questionnaire. Analysis of this data showed that the majority of the musical parameters significantly affected multiple dimensions of the imagined motions. The main conclusion is that decreasing musical parameters are strongly associated with descents, whereas increasing musical parameters are associated with increasing speed rather than ascent. A

surprising finding of this study is that musical-spatial analogies are often asymmetrical, as a musical change in one direction evokes a significantly stronger spatial analogy than its opposite. The study reported in this section is similar to Eitan and Zohar. However, instead of looking at imagined movement, the effect on physical hand movements is investigated. Similar tendencies in these two studies are expected.

For our study, we postulate the following hypotheses.

- H_1: *Between subjects there is a high amount of variation in types of motions patterns within stimuli.*
  A type of motion pattern is here defined as a family of gestures: a set of which the gestures look very similar. Because subjects will be unrestricted in their movements, a large variety of gestures are expected.

- H_2: *Within users there is a low amount of variation in types of motion patterns.*
  A preferred baseline motion is expected within a user.

- H_3: *There is a high amount of agreement between subjects within stimuli in higher order extracted feature derivatives.*
  Although the baseline motion between subjects is expected to differ, it is anticipated that feature derivatives[1] of those movements between subjects will postulate equal tendencies[2].

Based on Eitan's and Zohar's findings, we propose the following additional hypotheses regarding H_3:

- H_4: *Decreasing musical parameters are associated with descending motions.*
- H_5: *Increasing musical parameters are associated with ascending motions.*
- H_6: *Decreasing musical parameters are associated with decreasing speed.*
- H_7: *Increasing musical parameters are associated with increasing speed.*
- H_8: *Rhythmic variation has a large effect on change of direction.*
- H_9: *Subjects tend to move along with the beat.*
- H_{10}: *One-to-multi relations between musical parameters and motion features exist.*
- H_{11}: *The relations between musical parameters and motion features are asymmetrical* [3].

[1] Features such as verticality, speed orcurvature and derivatives such as minimum, maximum or average.

[2] For example, subject A performs a circular motion whereas subject B performs a vertical line movement. Hence the baseline motion differs. When looking at the derivative *average* of feature *speed*, H_3 predicts a similar pattern.

[3] A change in a musical parameter in one direction evokes a different size of change in a motion feature than its opposite.

5.2 Method

5.2.1 Participants

Thirty people participated in the experiment (15 females, 15 males, mean age = 22.9, St.Dev. = 1.7 years.). The group contained both non-musicians and musicians with varying amounts of musical experience. All participants were students at the Radboud University Nijmegen and participated on a voluntary basis. Four participants were left handed.

5.2.2 Materials

Hardware All experiments were conducted with a fixed setup. A solid stand with a variable height was positioned in the room. On this stand an Apple Cinema HD Display (23-inch LCD @ 1920 x 1200 pixels) was placed, connected to a Macbook Pro 13" 2010 model (2,4 GHz Intel Core Duo, 4GB 1067 MHz working memory). The height of the stand was adjusted such that the center of the screen was level with the eyes of the subject. Adjacent to this stand, on the left side of the screen (on the right for the perspective of the subjects), an height adjustable microphone stand was used with a Nintendo Wii remote controller attached to the end of arm. The arm end was pointed towards the participant, resulting of a parallel position of the Wii remote controller and the ground. The height of the microphone stand was adjusted such that the Wii Remote was level with the center of the Cinema Display. This relative position was determined optimal during pilot testing. A line mark on the ground was positioned in order to indicate where subjects were required to stand. Participants used one of our glove interfaces with two infrared LEDs, both situated on the index finger. Further, the diffuser module was present to ensure light transmission at different orientations. All subjects had to wear the glove on their right hand[4] The *(x,y)* position of the infrared LEDs were subsequently registered using the Wii remote's infrared camera.

[4]This was done to avoid that bodily constraints would affect their movements. For example, movement to the right is more easy with their right hand than with their left. This introduces a directional bias other than the result of variation in the independent stimuli.

Software A program for the experiment was written in Max/MSP (v. 5.1.7) [3] and Processing (v. 1.2.1) [18]. Processing provided the visual information of the experiment for the subject. Max/MSP provided the auditory stimuli and coordinated the timing for displaying events through the Processing framework. The position of the index finger was channeled from Max/MSP to Processing at 100Hz. The program OSCulator (v. 2.10.5) [52] provided the channeling of LED coordinates from the Wii remote controller to Max/MSP.

5.2.3 Stimuli

In total 18 auditory stimuli were created with Ableton Live (v. 8.2)., consisting of notes of equal length placed isochronous on a 1 bar segment in 4/4 time, played at 80BPM. Hence with a duration of 3 seconds each. They can be divided into two main groups and 2 special cases.

The first group consists of 3 pairs of musical contours of 8th note length. For each pair, a specific musical parameter was either increased or decreased over time while other musical parameters where held constant. The musical parameters varied were intensity, pitch and brightness[5].

The second group consists of rhythmic stimuli of note length 1/16, 1/8, 1/4 or 1/2. Other parameters were held constant. Another rhythmic stimuli contained an syncopated interval in order to simulate missed beat syncopation with 1/16 notes. Further, one stimulus of 1/8 notes was modified by adding accents on the first and fifth note.

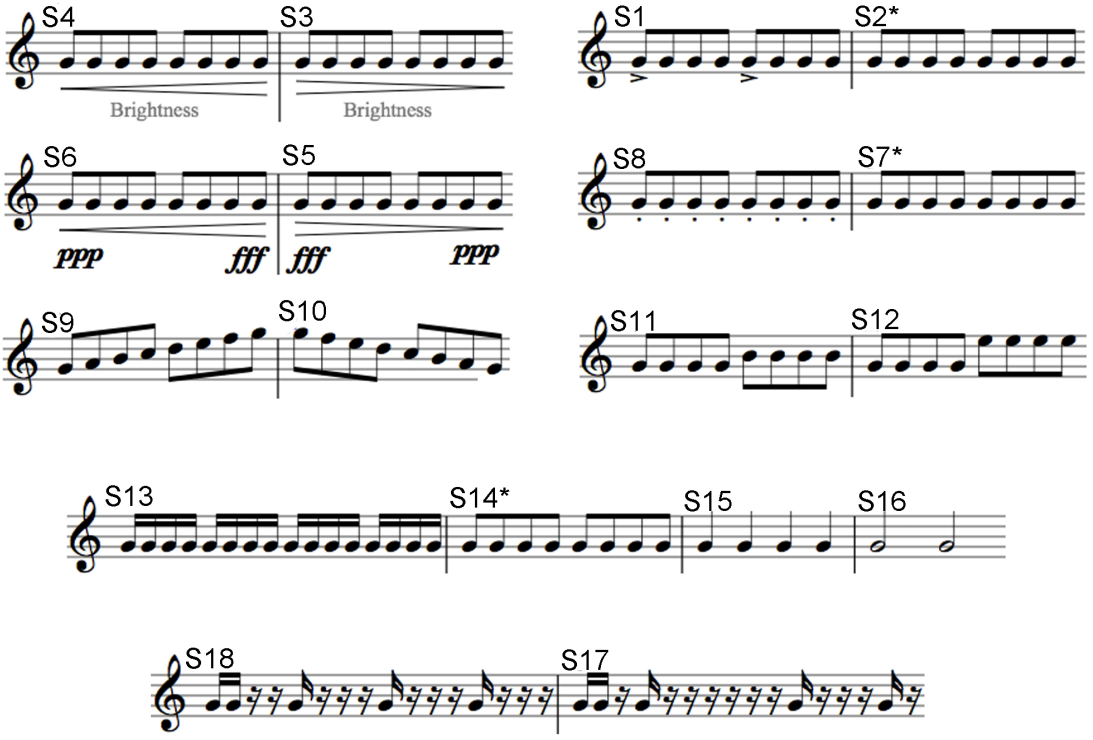
For the first special case, articulation was varied by adding staccato to all notes.

The second special case consists of pitch intervals. Two stimuli were created with a sudden change in pitch, one bigger than the other. The first 4 notes were equal in pitch whereas the last 4 notes were played either with a small increase or a high increase in pitch.

An overview of all stimuli can be found in Figure 5.5. Note that the starred numbered stimuli are equal.

[5]Brightness is a timbre feature of the sound. Increase of this parameter results in a brighter sound whereas decreasing results is a dull sound.

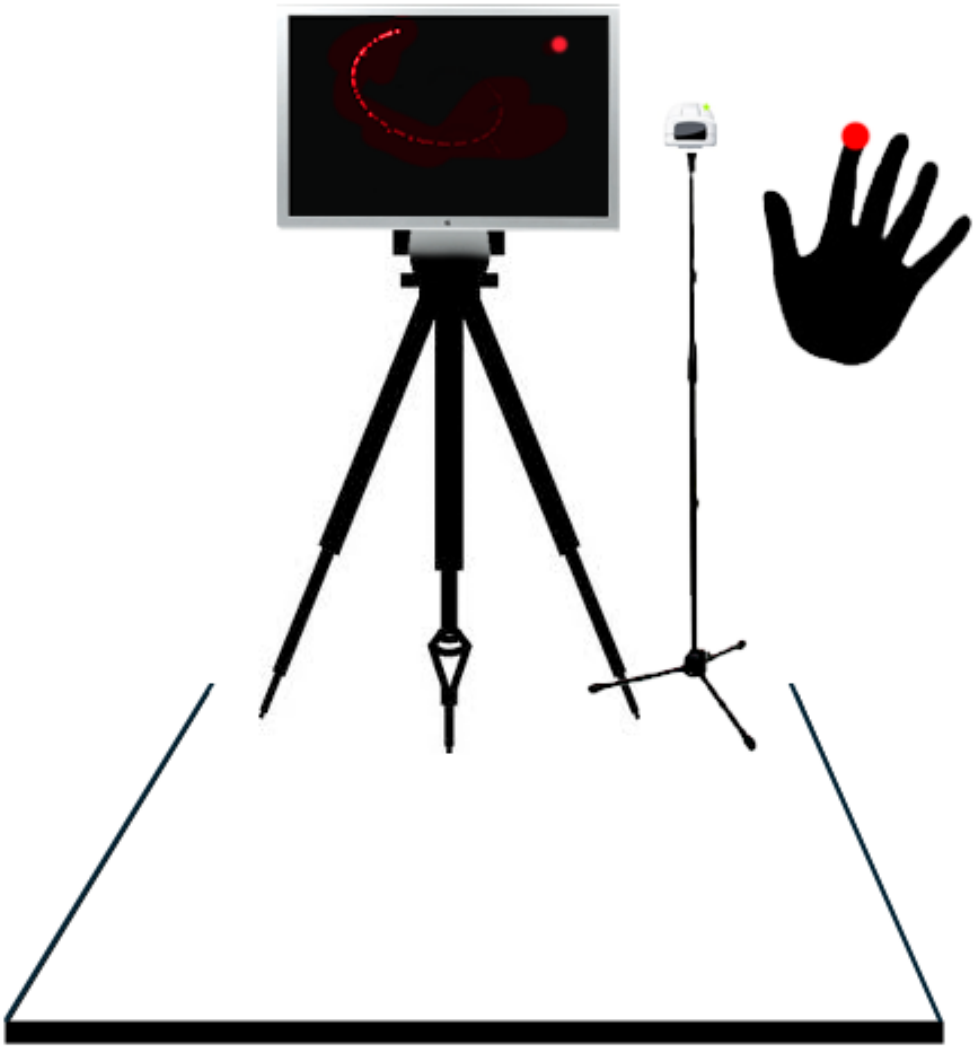
FIGURE 5.4: Overview of stimuli. S3, S4, S5, S6, S9 and S10 are musical contours. S3 And S4 are the brightness contours decreasing and increasing. S5 And S6 are the dynamics contours decreasing and increasing. S9 And S10 are the pitch contours increasing and decreasing. S1, S2, S7, S8, S11 and S12 are the special case stimuli. S1 And S2 are the accented and unaccented pair. S7* And S8 are the legato/staccato pair. S11 And S12 are the pitch intervals pair: small and large. S13, S14, S15, S16, S17, and S18 are the rhythmic stimuli. S13 Is the rhythmic sixteenth stimulus, S14* the rhythmic eighth, 15 the rhythmic quarter and 16 the rhythmic half. 17 And 18 are the syncopated and unsyncopated pair.



5.2.4 Procedure

Participants were situated in a noise free recording studio. They were asked to position their upper arm parallel to their bodies and their lower arms parallel to the ground, pointing with the index finger towards the Wii remote controller. Next their position was shifted forwards or backwards such that their right hand floated above the marked line. They were instructed to hold their right hand floating above the line in the orthogonal plane with this line during the experiment. The line on the ground was position parallel with the width of the screen on a distance of 1.5 meters. Thereafter the stand with the display was adjusted such that the bottom of the screen matched the height of their hands. Next the microphone stand with the Wii remote controller was adjusted to match the center height of the screen. Participants wore the interface glove on their right hand.

FIGURE 5.5: A graphical representation of the experimental setup.



Participants were allowed to familiarize themselves with the experimental setup. Their hand position was displayed as a small red circle (⌀ 10 pixels). Further, participants were able to see a fading trail of these circles representing the course of their hand movements. These trails were faded entirely after 1.5 seconds. A pre-experiment check ensured that left-handed participants could successfully make use of the glove. For this purpose a black screen was introduced with randomly appearing red dots (⌀ 30 pixels). Participants were asked to move their hand such that the movement circle hit the stationary red circle. This was repeated approximately 20 times until the experimenter and the participant together decided the participant was in control. Next the participant was asked to move to all four corners in the visual feedback. If there was no difficulty

detected or reported by a left-handed subject, the experiment continued to the next phase.

During this next phase, the participant was informed about the upcoming experimental procedure. Participants were presented with the following instructions. After each participant read the instructions, a summary was verbally repeated by the experimenter and questions could be answered. When everything was clear, the actual experiment commenced.

> **Instructions** *"During this experiment we will present sounds*[6]*. For each sound type there are two phases. Phase 1: Practice Phase. Phase 2: Recording Phase. During Practice Phase: Try to associate hand movement with the sound you hear. Draw this movement. When you have found an intuitive movement, select both red squares at the top. After selecting both squares, will go to the Recording Phase. During Recording Phase: You will see a countdown*[7]*. The sound starts playing at zero. Try to make the same movement as you decided was best in the previous phase. This phase is repeated once, so you can draw the same movement twice. After the two recording phases, two red squares appear at the bottom. You can now take a little break if necessary or select them both to go to a new sound. Press [spacebar] to start the experiment."*

A within subject design was chosen in which all participants completed all conditions in a randomized order. The actual experimental procedure follows the instructions above. During the recording and practice phases of a stimulus, coordinate data of the motion was sampled at 50Hz and saved to file. After the participants completed the experiment, they filled in a questionnaire that assessed personal information, musicality and their evaluation of the experiment and glove.

[6] Each sound represents one stimulus

[7] The countdown consisted of a succession of the numbers 4, 3, 2 and 1. The time between numbers was equal to the time between beats of the control stimulus, thus 3/8 seconds.

5.2.5 Data Analysis: Features

To capture higher order relationships in the data, a set of movements features were defined. These features were chosen to describe the primary motion's attributes. The following features were extracted: *horizontal position*, *vertical position*, *energy/speed*, *horizontal energy/speed*, *vertical energy/speed*, *curvature*, *direction*, *direction difference* and *running angle*. These features can be calculated real-time through Java in Max/MSP by using time windows of 2 successive coordinates. At a given time point i, features are defined as follows:

Horizontal Position : $HP = x_{t(i)}$

Vertical Position : $VP = y_{t(i)}$

Energy/Distance : $E = \sqrt{(x_{t(i+1)} - x_{t(i)})^2 + (x_{t(i+1)} - x_{t(i)})^2}$

Horizontal Energy : $HE = x_{t(i+1)} - x_{t(i)}$

Vertical Energy : $VE = x_{t(i+1)} - x_{t(i)}$

Curvature : $Curvature = \frac{i}{R}$, where

R[8] $= \sqrt{(x_{t(i)} - CX)^2 + (y_{t(i)} - CY)^2}$, where

CY[9] $= (\frac{-1}{S1}) * (CX - \frac{x_{t(i)} + x_{t(i+12)}}{i+2}) + (\frac{y_{t(i)} + y_{t(i+12)}}{i+1})$,

CX[10] $= \frac{S1*S2*(y_{t(i)} - y_{t(i+24)}) + S2*(x_{t(i)} - x_{t(i+12)}) - S1*(x_{t(i+12)} - x_{t(i+24)})}{2*(S2-S1)}$,

$S1$[11] $= \frac{(y_{t(i+12)} - y_{t(i)})}{(x_{t(i+12)} - x_{t(i)})}$,

$S2$[12] $= \frac{(y_{t(i+24)} - y_{t(i+12)})}{(x_{t(i+24)} - x_{t(i+12)})}$

Direction : $D = atan2(x_{t(i)} - x_{t(i+1)}, y_{t(i)} - y_{t(i+1)})$

Change in Direction : DC[13] $= D_t(1) - D_t(2)$

Running Angle : RA[14] $= atan(\frac{y_{t(2)} - y_{t(1)}}{x_{t(2)} - x_{t(1)}})$

[8] Radius of intersecting circle

[9] The y coordinate of the center of the intersecting circle.

[10] The x coordinate of the center of the intersecting circle.

[11] Slope of first en second coordinate

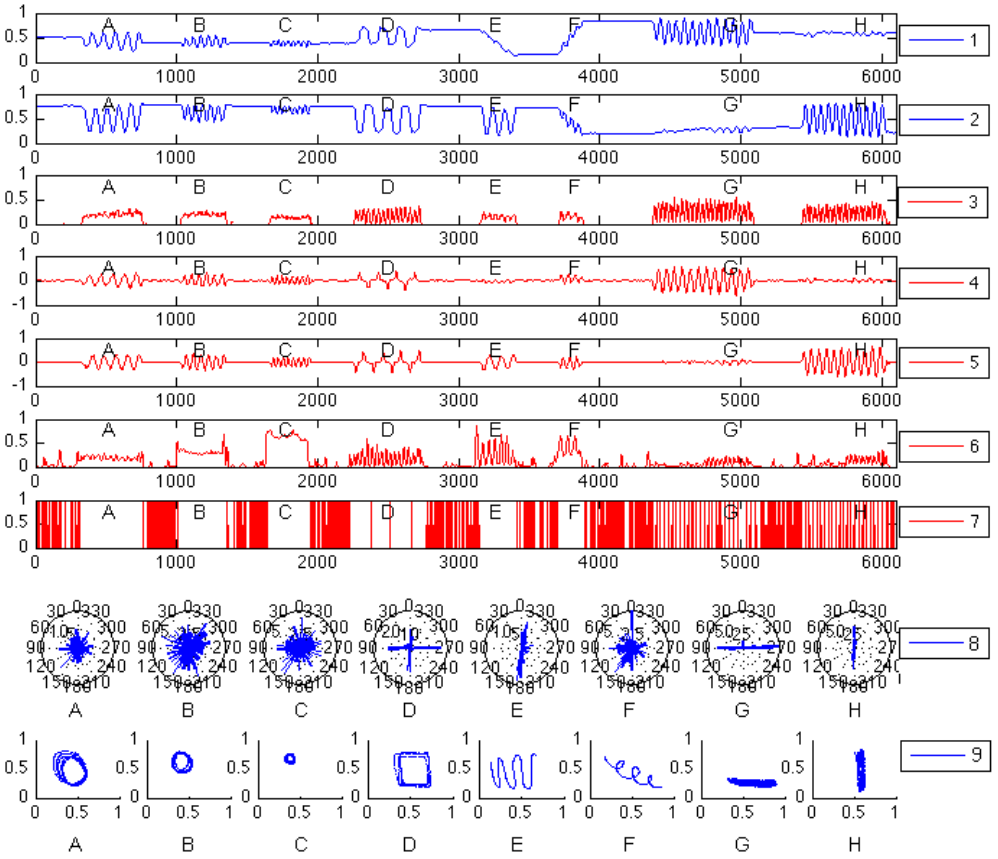
[12] Slope of second en third coordinate

[13] Since the direction function outputs values between 0 and 2π where 0 and 2π are equal directions, an additional computation has been performed in order to obtain DC. Without this computation, the actual difference between two directions in a movement can be very small, but as seen by the direction difference function as very large. For example, the difference between an angle of 2π radians and 0 radians equals 0 and not 2π. The computation eliminates this error by 'connecting' 0 and 2π as-if they were circular, counting beyond 2π by starting at 0. Since this introduces 2 possible solutions for each pair of input, the minimum solution is chosen which represents the actual direction difference.

[14] The running angle is defined as the relative angle of a vector with respect to the horizontal plane.

Feature Validation Features were validated by performing exemplar gestures and evaluating the feature output. Results from these investigations are show in Figure 5.6.

FIGURE 5.6: Feature outputs over time. In Subfigure 9, the hand motions A to H are shown. They were drawn sequentially with breaks. This resulted in feature values over time, shown in Subfigures 1 to 8. Where in time the hand motions occur in time is indicated by the corresponding letter. Subfigure 1 represents the Horizontal Position feature. Subfigure 2 represents the Vertical Position feature. Subfigure 3 represents the Energy/Speed feature. Subfigure 4 represents the Horizontal Energy feature. Subfigure 5 represents the Vertical Energy feature. Subfigure 6 represents the Curvature feature. Subfigure 7 represents a thresholded Direction Difference feature where a high difference in direction represents a 'beat' in a movement. Subfigure 8 represents the Direction feature, visualized in a polar plot.



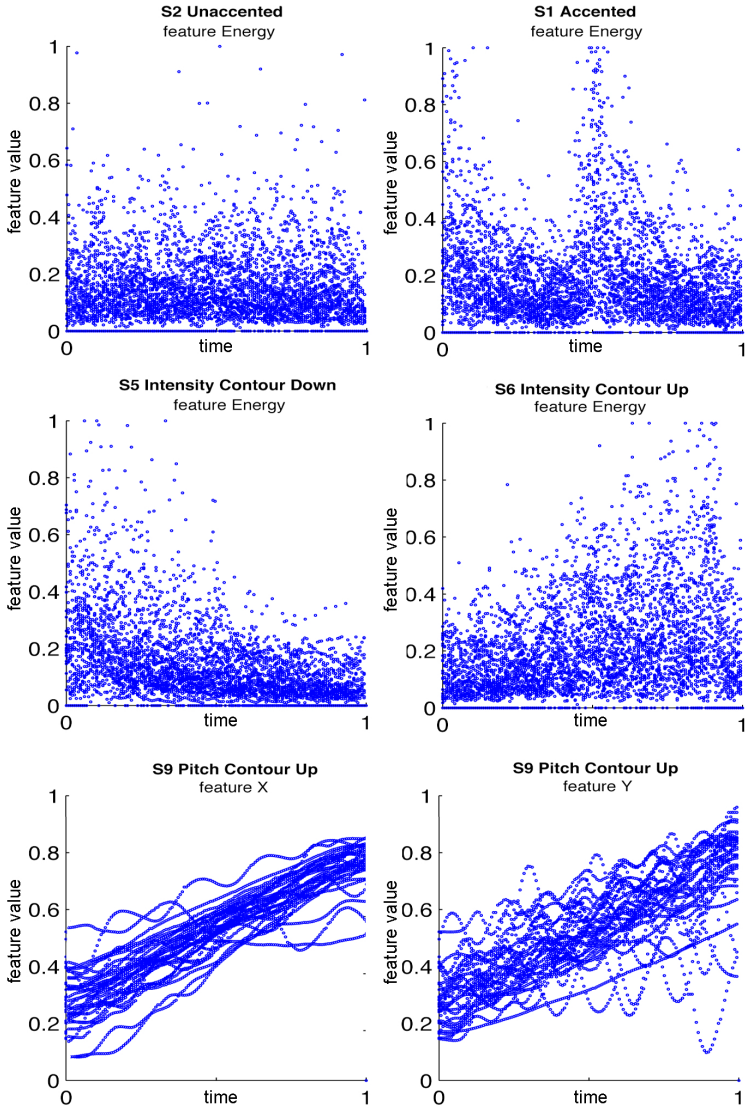
Feature Derivatives Derivatives from features are also calculated in order to summarize a set of values over time to a singular value per motion. They were chosen to cover general descriptive aspects of a feature. The derivatives per feature are: minimum, maximum, standard deviation, average. For the X and Y features, also the start-end differences are calculated.

5.3 Quantitative Results

5.3.1 Feature Distributions

These initial analysis focusses on feature distributions which display between-user trends in our data. In Figure 5.7 three examples are given. The top two figures display a difference in energy distribution for the accented an unaccented stimuli at points 0 and 0.5 relative in the stimulus's playback in time. The middle two figures show a decreasing and increasing tendency of the Energy feature in the when the intensity of the sound increases and decreases respectively. The bottom two figures indicate the trends in listeners to move to the right and upwards when the pitch is increased. These trends are subsequently verified using statistical tests (see Table 5.2).

FIGURE 5.7: Examples of feature distributions.



5.3.2 Statistical Report: Control Stimuli Comparisons

MANOVAs: A series of MANOVAs were performed in order to detect differences in derivatives of features between a stimulus with a varied musical parameter and the control stimuli without varied parameters.

Independent variables

- **Stimulus:** Varied{S1‖ S3 ‖ S4 ‖ S5 ‖ S6 ‖ S8 ‖ S9 ‖ S10 ‖ S11 ‖ S12 ‖ S13 ‖ S15 ‖ S16 ‖ S17 ‖ S18} , Control{S2 ∧ S7 ∧ S14}

Dependent variables

- **Feature X:** D_1, D_2, D_3, D_4, D_5.
- **Feature Y:** D_1, D_2, D_3, D_4, D_5.
- **Feature Energy:** D_3, D_4, D_5.
- **Feature Horizontal Energy:** D_3, D_4, D_5.
- **Feature Vertical Energy:** D_3, D_4, D_5.
- **Feature Curvature:** D_3, D_4, D_5.
- **Feature Direction:** D_3, D_4, D_5.
- **Feature Direction Change:** D_3, D_4, D_5.
- **Feature Running Angle:** D_3, D_4, D_5.

Where
D_1 = start-end,
D_2 = average,
D_3 = standard deviation,
D_4 = maximum,
D_5 = minimum.

A summary of effects of the analyses is shown in Table 5.1. First the F(37,82) values of the MANOVAs on the Stimulus versus Control on Feature Derivatives are presented. Next the follow-up ANOVAs per Feature Derivative are presented with the F(1,118) values. Significance levels are indicated with asterisks.

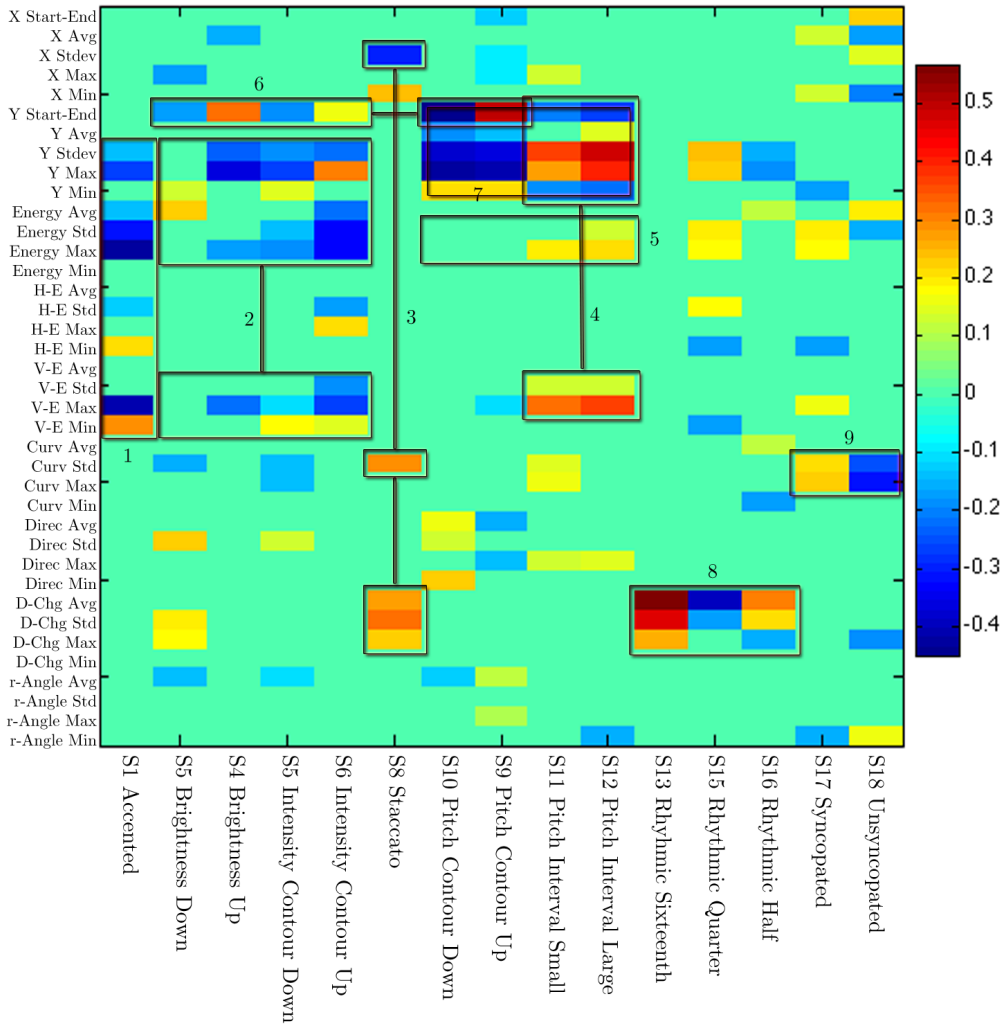
Discriminant Analyses: The multivariate tests evaluate the correlation between dependent variables and therefore are strong in detecting group differences. Follow-up ANOVAs are then performed to assess which dependent variables contribute significantly to this differences. The univariate tests omit interaction effects. To take this into account, follow-up discriminant function analyses are performed.

TABLE 5.1: Summary of Analyses

	Condition (stimulus number vs. control)														
	S1	S3	S4	S5	S6	S8	S9	S10	S11	S12	S13	S15	S16	S17	S18
MANOVA	5.53 ***	4.05 ***	3.96 ***	8.16 ***	5.92 ***	1.49 m	10.58 ***	4.85 ***	4.81 ***	4.18 ***	3.53 ***	2.68 ***	7.44 ***	4.37 ***	3.49 ***
Follow-up ANOVAs															
X Start-End							9.19 **								9.35 **
X Avg			5.01 *											4.05 *	5.57 *
X Stdev						5.71 *	5.61 *								4.15 *
X Max		6.89 **					5.52 *		4.02 *						
X Min						4.75 *									
Y Start-End		6.69 *	22.73 ***	15.63 ***	7.58 **		143.13 ***	52.28 ***	10.18 **	16.64 ***				4.35 *	
Y Avg							11.32 ***	9.19 **		4.52 *					
Y Stdev	6.34 *		11.82 ***	15.23 ***	15.85 ***		72.79 ***	35.46 ***	34.27 ***	50.22 ***		8.52 **	9.54 **		
Y Max	20.21 ***		26.60 ***	31.38 ***	30.75 ***		94.29 ***	48.11 ***	20.21 ***	35.14 ***		6.87 **	13.22 ***		
Y Min		4.04 *		9.03 **			24.25 ***	12.20 ***	8.45 **	10.48 **				6.36 *	
Energy Avg	6.02 *	10.86 ***			15.92 ***								5.39 **		6.83 **
Energy Std	28.14 ***				34.85 ***					4.02 *		5.07 *		8.78 **	4.25 *
Energy Max	53.24 ***		6.78 **		35.90 ***				9.68 **	10.55 **		4.62 *		7.37 **	
Energy Min															
H-E Avg															
H-E Std	4.06 *				8.70 **							4.20 *			
H-E Max															
H-E Min	13.55 ***				13.21 ***							4.05 *		6.68 *	
V-E Avg															
V-E Std	20.28 ***				11.50 ***				4.32 ***						
V-E Max	50.12 ***		9.43 **	5.05 *	23.53 ***		6.73 *		27.58 ***					6.59 *	
V-E Min	24.26 ***			14.06 ***	6.44 *							4.08 *			
Curv Avg													4.76 *		
Curv Std						6.78 **			5.71 *					10.53 **	11.53 ***
Curv Max									6.65 *					11.92 ***	17.68 ***
Curv Min													12.63 ***		
Direc Avg							13.69 ***	6.59 *							
Direc Std				6.86 **				4.77 *							
Direc Max							7.35 **	14.06 ***	4.33 *	5.23 *					
Direc Min															
D-Chg Avg						5.96 *					60.60 ***	22.33 ***	37.67 ***		
D-Chg Std		7.99 **				8.22 **					39.25 ***	4.25 *	18.43 ***		
D-Chg Max		6.86 **				4.00 *					12.12 ***		9.94 **		6.84 **
D-Chg Min															
r-Angle Avg		3.94 *		5.02 *			6.52 *	4.24 *							
r-Angle Std															
r-Angle Max							4.71 *								
r-Angle Min										4.86 *				5.35 *	4.95 *

m: p<0.07, * p<0.05, ** p<0.01, *** p<0.001

FIGURE 5.8: Summary table of discriminant analyses: effect sizes and direction per feature derivative in stimulus versus control. Boxes indicate relevant findings (see Section 5.5).



5.3.3 Statistical Report: Pairwise Comparisons

MANOVAs: 5 Separate MANOVAs were performed in order to detect differences in derivatives of features between a stimulus and its corresponding opposing pair.

Independent variables

- Brightness Contour(S2,S3), Intensity Contour(S5,S6), Pitch Contour(S10, S9), Pitch Interval(S11,S12), Syncopation(S17,S18)

Dependent variables

- **Feature X:** D_1, D_2, D_3, D_4, D_5.
- **Feature Y:** D_1, D_2, D_3, D_4, D_5.
- **Feature Energy:** D_3, D_4, D_5.

- **Feature Horizontal Energy:** D_3, D_4, D_5.
- **Feature Vertical Energy:** D_3, D_4, D_5.
- **Feature Curvature:** D_3, D_4, D_5.
- **Feature Direction:** D_3, D_4, D_5.
- **Feature Direction Change:** D_3, D_4, D_5.
- **Feature Running Angle:** D_3, D_4, D_5.

Where
D_1 = start-end,
D_2 = average,
D_3 = standard deviation,
D_4 = maximum,
D_5 = minimum.

A summary of effects of the analyses is shown in Table 5.2. The F(37,22) values of the MANOVAs and F(1,118) values of the follow-up ANOVAs are presented. Significance levels are indicated with asterisks. In Figure 5.9 a summary of the discriminant analyses is presented.

5.3.4 Beat Synchronization

In order to find evidence for beat synchronization [51], frequency analyses were performed for each subjects' energy level per rhythmic stimulus and for all 8th note stimuli. In Figure 5.10 and Figure 5.11 the results are plotted.

5.4 Qualitative Results

5.4.1 Subjective Motion Interpretations

In Appendix A some example motions per participant per stimuli are depicted. The starting positions are marked with a red circle. Visual inspection of these data suggests between participants that there is low variance in types of motion patterns. Further, participants tend to agree (n=28) to often move along the beat by making small hills per beat. Alternately, they create a circular movement per beat. Other shapes are uncommon. Another observation is that participants move primarily from left to right but do not use all space available. Participants tend to center their movements, both horizontally and vertically. Their hand position is near the center of the screen halfway during a gesture. Participants also tend to move upwards and downwards respectively when a musical parameter is increased or decreased.

TABLE 5.2: Summary of Analyses

	Condition				
	S3 S4	S5 S6	S10 S9	S11 S12	S17 S18
MANOVA					
	3.82 ***	5.49 ***	7.29 ***	0.87	1.02

Follow-up ANOVAs	S3 S4	S5 S6	S10 S9	S11 S12	S17 S18		S3 S4	S5 S6	S10 S9	S11 S12	S17 S18
X Start-End			8.83 **			V-E Std	7.26 **				
X Avg						V-E Max	4.30 **		12.21 ***		
X Stdev						V-E Min					
X Max						Curv Avg					
X Min						Curv Std		10.80 **			
Y Start-End	22.89 ***	18.39 ***	150.27 ***			Curv Max		10.17 **			
Y Avg						Curv Min			8.65 **		
Y Stdev			4.71 *			Direc Avg			23.04 ***		
Y Max	7.38 **		7.68 **			Direc Std	5.95 *	14.17 ***			
Y Min						Direc Max	4.06 *		16.26 ***		
Energy Avg	17.62 ***	15.21 ***				Direc Min		5.98 *			
Energy Std	9.76 **	5.24 *				D-Chg Avg					
Energy Max	5.51 *					D-Chg Std	6.39 **				
Energy Min						D-Chg Max	5.13 *				
H-E Avg						D-Chg Min					
H-E Std	4.74 *		4.78 *			r-Angle Avg	12.52 ***	6.69 *	17.51 ***		
H-E Max						r-Angle Std					
H-E Min						r-Angle Max					
V-E Avg			4.90 *			r-Angle Min					

m: p<0.07, * p<0.05, ** p<0.01, *** p<0.001

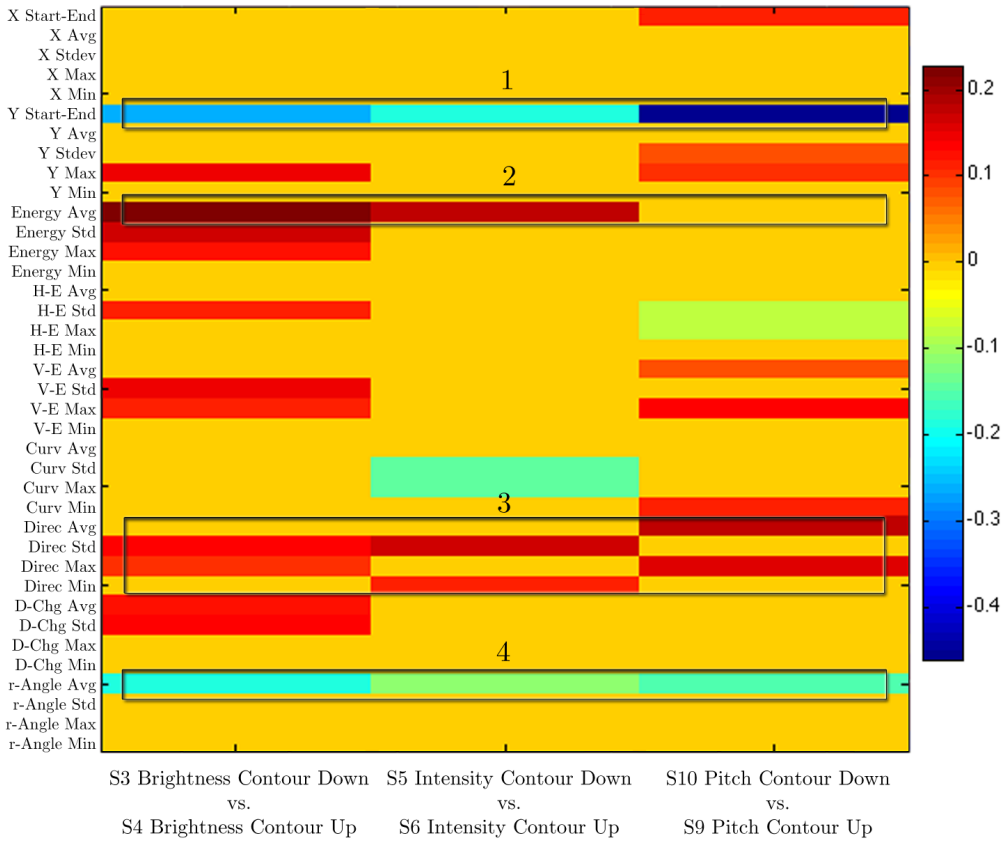
20 Out of 30 participants make a near equal motion in one or both Pitch Interval stimuli. This motion starts bottom left and moves horizontally. After a sudden pitch increase, the motion also moves suddenly to a higher level. The amount of vertical increase depends on the strength of the pitch increase.

For the control stimuli S2, S7 and S14, 7 participants made 3 different types of movements, 15 participants made 2 different types of movements and 8 subjects make 1 type of movement. Hence, 77% of the participants made two or three similar types of movements while 23% made differing types of movements.

Furthermore, 10 participants made a single type of movement for all stimuli whereas 14 participants made 2 types of movements.

Participant 10 is a possible outlier given the repetition of gestures. However, this subject conforms to H_2 and could add to the between subject similarity as posed in H_3.

FIGURE 5.9: Summary table of discriminant analyses: effect sizes and direction per feature derivative in stimulus contour up versus stimulus contour down. Boxes indicate relevant findings.



In order to validate these subjective claims, statistical tests could quantify these findings.

5.4.2 Questionnaire Results

On a 1 to 5 point scale, where higher implies a better evaluation, subject's replied to 5 questions regarding the experiment and the interface.

To the question assessing the difficulty of associating sounds with movements, subjects replied positive with 3.93 out of 5 points with a standard deviation of 1.11 points. This indicates that most subjects found it easy, or as the standard deviation indicates, at least without any problems or even very easy to associate sounds with movements.

To the question assessing the difficulty of performing movements, subjects replied positive with 4.03 out of 5 points with a standard deviation of 0.99 points. This indicates that most subjects found it easy, or as the standard deviation indicates, at least without any problems or even very easy to perform these movements.

FIGURE 5.10: Frequency analysis of rhythmic stimuli. The red lines indicate individual results, the blue line the average of all subjects.

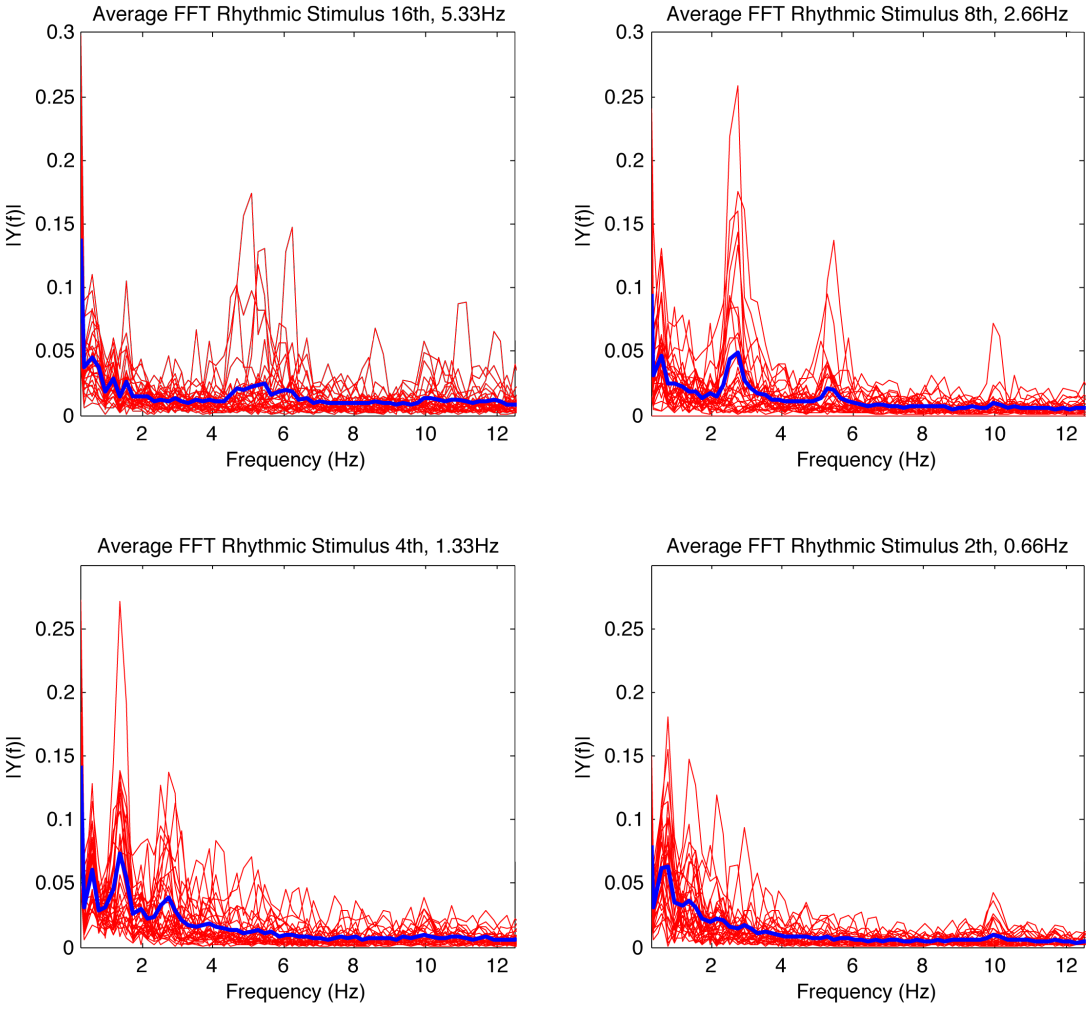
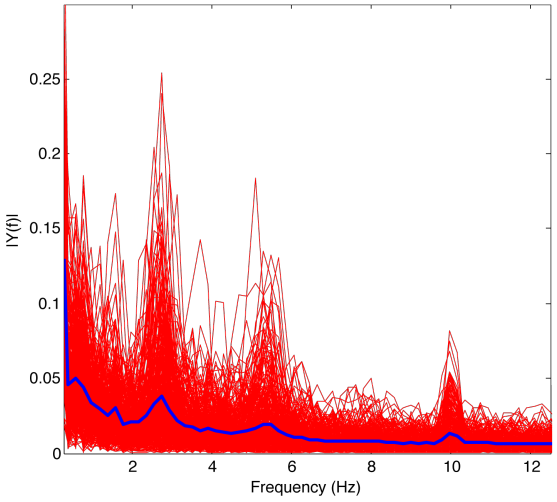


FIGURE 5.11: Frequency analysis of all stimuli with 1/8 notes. The red lines indicate individual results, the blue line the average of all subjects.



To the question assessing the experience of latency of performed movements and visual feedback, subjects replied positive with 4.10 out of 5 points with a standard deviation of 0.99 points. This indicates that most subjects found that there was a low amount, or as the standard deviation indicates, at least no disturbing experience of lag or even no lag at all while performing the movements.

To the question assessing the comfortability of the glove, subjects replied positive with 3.87 out of 5 points with a standard deviation of 1.13 points. This indicates that most subjects found the glove comfortable to wear, or as the standard deviation indicates, at least without discomfort or even very comfortable to wear.

To the question assessing the physical tiredness of the hand and/or arms, subjects replied positive with 3.98 out of 5 points with a standard deviation of 1.04 points. Thus this indicates that most subjects found that performing was not tiring, or as the standard deviation indicates, at least no disturbing experience of tiredness or even no tiredness of the hands and/or arms at all while performing the movements.

Overall subjects were very positive with an average of 3.98 out of 5 points per question with a standard deviation of 0.09 points. Other feedback subjects left to report experiences, suggestions, improvements or other thoughts, were as follows.

- *"In particular the slow motions felt good, for there I felt I could keep up with the speed while staying accurate."*
- *"I liked it, but the lower corners were hard to get."*
- *"Use less boring sounds."*
- *"The visual feedback might have had influence on the movement you selected. A concatenation of points lets you make one smooth movement."*
- *"The visual feedback caused me to perform different movements by focussing more on shapes and large movements. No visual feedback would direct the attention more to the body en perhaps help to associate movement better with sound."*
- *"I heard one particular sound multiple times, though I'm unsure."*
- *"The latency makes it hard to perform movements."*
- *"Please continue developing this product and make it affordable for buyers!."*
- *"The abnormal rhythms were very hard to perform."*
- *"I had a movement strategy, however I changed it during the course of the experiment."*

5.5 Discussion

Against hypothesis H_1, results show there is is low amount of variation between subjects in types of motion patterns. Indicated by the qualitative results, the primary tendency of the participants was to move from left to right while making a hopping type of motion. Further, the most common motion is either horizontal, linearly upward or linearly downward. On the other hand, within participants there is a low amount of variation in types of motion patterns used. This in favor of hypothesis H_2. Subjective motion interpretation confirms that participants preferred one or two baseline motions, of which they altered the attributes. However, statistical tests must yet confirm this finding by providing a quantified result. This remains future research.

According to H_3, higher order features extracted from motion patterns show similar tendencies, independent from the motion type. The statistical analyses confirm this: there are significant differences between derivative values of motional features of the varied stimuli and the control stimuli. H_{4-11} specify these tendencies. Regarding these hypotheses, Figure 5.8 and Figure 5.9 provide the results. Box 6 in the former figure and Box 1 in the latter, show a positive and negative significant start-end difference when a musical parameter is increased or decreased respectively. This in favor of H_{4+5}. Decreasing musical parameters are associated with descending motions whereas increasing musical parameters are associated with ascending motions. This partially confirm the results of Eitan and Granot [16]. Their findings indicate that decreasing musical parameters are strongly associated with descents, whereas increasing musical parameters are associated with increasing speed rather than ascent.

When looking at the Increasing and decreasing musical parameters, they are associated with increasing and decreasing and increasing speed/energy respectively. Box 2 in Figure 5.8 also shows that this does not hold for the pitch parameter. Further, solely the vertical component in the energy parameter is affected. This partially confirms H_{6+7}.

Directional change is significantly affected by rhythm, confirming H_8. In Box 8 in Figure 5.8 this effect is highlighted. Furthermore, there appears to be periodicity in the participants' movements. In Figure 5.10 the frequency distributions for the rhythmic stimuli are plotted. For each rhythm, peaks in the power spectrum are present around the expected frequency. For example, for the rhythmic 16th stimuli, peaks are present at $\frac{16beats}{3seconds} = 5.33$ Hz. In Figure 5.11 the frequency distributions of all stimuli with 8th isochronous notes is displayed. The expected peaks are present at 2.66 Hz. This confirms H_9: participants move along with the beat. Note that the

average peak appears lower by peak cancellation due to slight differences of motion synchrony. Further, other peaks are present at a factor of the expected frequency due to periodic repetitions. This supports the findings of Toiviainen and Luck [51] who found that periodicities are present in music induced movement.

According to Figure 5.8 not a single motion parameter is solely effected by one musical parameter. Moreover, multiple musical parameters affect multiple motional features. Hence multi-to-multi relations exist, partially confirming H_{10} and the findings of Eitan and Granot [16].

In Figure 5.9 it becomes apparent that asymmetry is present in some of the effect sizes between opposing pairs of stimuli. Significant effects should be absent when asymmetry occurs due to equal feature derivatives. Therefore we conclude that the relations between musical parameters and motion features are partially asymmetrical. For example, the energy feature is asymmetrical effected by brightness and intensity, but not by pitch (see Box 2).

An unforeseen tendency is the effect of accenting on the energy levels, specifically the vertical component of the energy. This is indicated by the significant increase of maximum and standard deviation of Y and maximum energy levels, as depicted in Box 1 in Figure 5.8.

When a sound is played in staccato, this effects increase changes in direction and curvature (see Box 3, Figure 5.8). This could indicate that subjects have made a lot of shaped gestures, rather than hopping lines. Cross checking with the qualitative data, this is confirmed.

Box 4 in Figure 5.8 shows the effect of changes in musical parameters in different sizes on the motional features. It is shown that a larger increase of such a parameter produces a larger effect size on the motional features.

Syncopation effects primarily the curvature of a motion. Box 9 in Figure 5.8 displays this effect. Multiple participants reported the following: *"The abnormal rhythms were very hard to perform."*. Cross-checking with the raw motion patterns, participants create complex forms, explaining these curvature values. Hence, a complex rhythm is associated with more complex motions.

5.6 Conclusion

Our study provides a detailed view on the effects of musical parameters on hand motions by evaluating their physical characteristics. Remarkably, participants highly agreed in the type of motions associated with sounds. Further they produced a low amount of variation in types of motions individually, even though they were unrestricted. However, the suggestive evidence requires future research to quantify this finding.

Our findings partially confirm the results of Eitan and Granot [16]. Their findings indicate that decreasing musical parameters are strongly associated with descents, whereas increasing musical parameters are associated with increasing speed rather than ascent. The former is confirmed by the findings in our study, however for the latter contradictory evidence was produced. Our study indicates that reverse holds, increasing musical parameters are associated with ascent rather than increasing speed. Furthermore, Fourier analysis provided supporting evidence for findings of Toiviainen and Luck [51] and Dahl [12] is found, suggesting that body movements synchronize with the periodic structure in music.

Evident is that tendencies between musical changes and motional features exist, though the relations are complex in nature. Multi-to-multi mappings are present and not all effects are symmetrical. The associations and tendencies found with this study, provide coherent knowledge for creating a solid musical mapping for the hand gesture driven interface (see Chapter 6). Moreover, it provides further support for the musical embodied cognition thesis [35]. The tendencies in the results indicate that people have internalized abstract representations of sound generating movements. This might be a culturally shared representation of abstract sound features, directly linked to movement. Hence this provides a coupling between perception-action as postulated by the musical embodied cognition thesis.

Chapter 6

Towards Sound Production

In this chapter it is explained how the hand gestural information, existing out of of analytical control gestures (see Chapter 4) and continuous motional feature values (see Chapter 5), can be used for sound production. Furthermore a developed proof-of-concept DMI is described.

The musical possibilities are to some extend restricted by the glove interface. However, the control gestures still provide a large possibility space. The two gestural types of information provide different types of control. First, the discrete nature of the analytical control gestures provide trigger signals for musical events. These events could encompass functionalities like *play*, *pause*, *next instrument*, *previous filter* or *enable loop*. The 'pinching' functionality (see Section 3.3.2) can be used to determine the start and ending of an analytical control gesture. After the classifier component has recognized such a gesture with enough certainty, the trigger signal is sent to a sound generating component, which consequently initiates a musical event. In Figure 6.1 they are represented in Max MSP at the top left. The second type of gestural information contains the continuous motional feature values, or 'holistic' gestures. As described in Section 2.2.2, these are not actual gestural motions but higher order derivatives of the gesture made. These motion properties such as *speed* and *curvature* are continually updated when the user is making a hand movement. In Figure 6.1 they are represented in Max MSP at the top right as sliders. These continuous values can be coupled to musical parameters, such as *pitch* or *loudness*. Both types of control gestures connect to a sound generating component (see bottom 6.1), which determines how the control gestures are mapped to sound. This musical mapping for each control value can be achieved in 4 ways, described by the possible connections between motional parameters and musical parameters. They are summarized in Figure 6.2.

FIGURE 6.1: Outputs from the interface inside Max MSP. 17 Analytical control gestures (top left) provide a sound module (bottom) with a signal via red lines. 7 Holistic control gestures (top right) provide a continuous signal to via blue lines. In the image, gesture 4 is triggered and sliders represent the values of the holistic gestures.

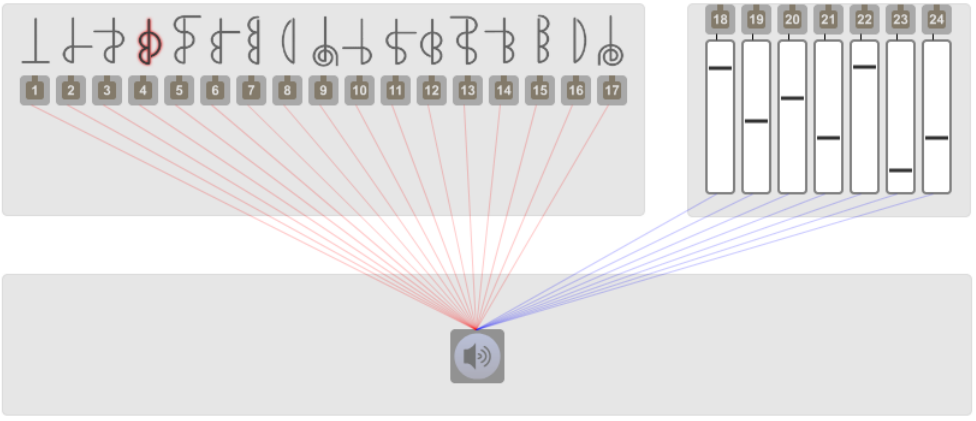
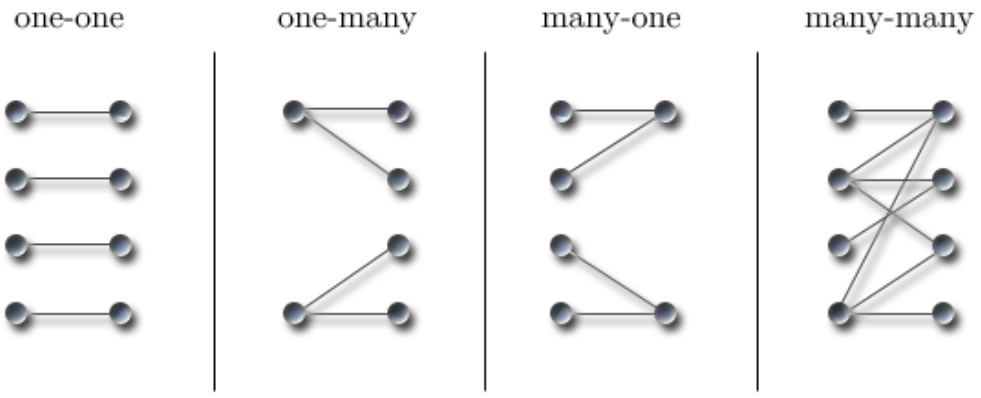


FIGURE 6.2: Possible types of mappings from motion parameters (left dots) to musical parameters (right dots).

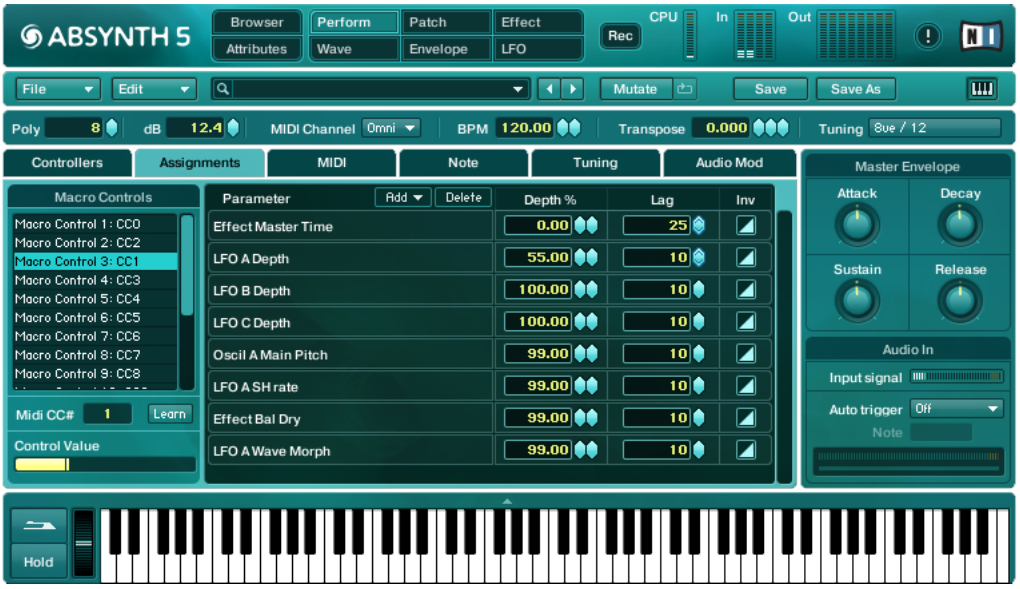


According to a study of Hunt and Kirk [25], the many-to-many mappings are preferred by users since it provides a more challenging and richer experience compared to one-to-one mappings. This finding is further supported by results from our music-motion-study (see Chapter 5), which concluded that many-to-many mappings from musical parameters to motion parameters were associated within users. Apparently listeners prefer to use more complex moving strategies, which makes it plausible they also prefer this as a performer. These findings suggest that for a musical mapping, a many-to-many approach should be adhered to.

A musical mapping can be realized internally or externally of Max Msp. Internally, the control gestures can be transformed to an audio signal. Externally, the information can be transmitted via MIDI to control other digital instruments.

An example of such an external sound producing system is the software synthesizer Absynth 5 [26]. In Figure 6.3 its graphical user interface is depicted. The synthesizer's parameters can be controlled via MIDI channels. In this manner, it can be controlled by the gestural information of our DMI. The latter is used in our prototype, which will be described in the next section.

FIGURE 6.3: GUI of Absynth's from Native Instruments.



6.1 DMI Prototype

As a proof of concept, we created a DMI prototype using our interface and the preset *'arcadia repeats'* in Absynth 5. One-to-one and one-to-many musical mappings were chosen. The gestural feature value of *horizontality* was mapped to pitch, whereas *horizontality* was mapped to the 8 synthesizer parameters displayed in Figure 6.3. The pinch property of the interface was used as a switch to mute/unmute the performance. The visual and auditory results can be viewed at:

`http://vimeo.com/rbrth/dmi-prototype-demo`

For this prototype the gesture recognition functionality was omitted. However an example without auditory events can be viewed at:

`http://vimeo.com/rbrth/gesture-recognition-demo`

Chapter 7

Conclusion

This thesis described the creation of a hand gestural controlled digital musical instrument. Using an iterative design methodology, we explored different glove designs. The final glove design features two diffused infrared light sources at the fingertips, which can be tracked at every angle by the Nintendo Wii remote controller up to a range of 2.30 meters. This enables users to have a stretched arm in any direction while performing while the movement resolution remains high (1.90-2.54 mm/px). The hardware required for this setup is affordable ($<$ 20 €), fast ($<$ 50 ms) and has a high sampling rate (μ 98 Hz). Moreover, the system is portable (0.2 dm^3) and easy to replicate within an hour. According to a user study, part of Chapter 5, the glove is comfortable to wear and gestures can be easily performed with it.

On the software side we merged predefined software frameworks to i) obtain a system which could acquisition hand generated control signals and ii) transform and output these signals to auditory information and iii) display the gestural information as visual feedback. The first functionality is achieved by OSCulator [52], sending the data via OpenSoundControl towards Max MSP [3]. The latter component takes care of the second functionality. This framework is a visual programming environment especially designed for musical productions. It can transform the gestural data to auditory data and output it as either an audio signal or as MIDI data. Furthermore it forwards the gestural information to Processing via UDP, which provides the third functionality. Users reported (see Chapter 5) that software provides a smooth visual feedback without disturbing delay. An important final design feature in the software merges tracked blobs together when they are close. This enables the user to 'pinch', adding an extra dimension in the interaction.

The interface is capable of interpreting 2 types of motional information: analytical and holistic control gestures. The first are like symbols with a binary existence. They can act as buttons or switches. The second type of motion information is

based on features derived in real-time from the gestures. This continuous stream of higher order data can act for direct control of musical parameters.

In Chapter 4 the development and analysis of a set of analytical control gestures was explored. In total 17 unistroke symbols were designed, inspired on the musical gestures made by conductors. A Hidden Markov Model was included in Max MSP to classify these symbols. This classifier was especially designed to recognize gestures made with a low variance between samples [8], in which it is known that musicians are proficient [46, 47]. Moreover, the model is trainable with only a single example. After multiple gesture data collections, simulations were run in order to determine to what extend the classifier was performing. After a few hours of training, users can achieve a recognition rate above 99.15 % by using only a singular training example. However, this did not hold for all participants. The performance of the classifier was compared to a second KNN classifier. Results indicated that for all subjects a performance above 99% could be reached. Hence indicating further efforts are needed to improve the HMM classifier.

In Chapter 5 the associations of listeners between musical parameters and holistic control gestures were investigated. The chapter reported on an experiment that was conducted to measure the effect of changes in dynamics, pitch, brightness, articulation, syncopation or rhythm on hand motions. This was measured by derivatives of motional features. Results indicated that many features of the motions are significantly affected by many musical parameters. Furthermore, there is suggestive evidence that between and within participants there exists a low amount of motional type variance. These results will provide essential knowledge to create an intuitive musical mapping from hand motions to sounds, as described in Chapter 6. Moreover, supporting the musical embodied cognition thesis, it suggests that people have an internalized abstract representation of sound generating movements: a culturally shared representation of abstract sound features directly linked to movement. The results confirm findings in earlier studies [12, 16, 51]

Finally in Chapter 6 it is explained how the hand gestural information can be used for sound production. Multiple types of musical mappings are available. According to our study in Chapter 5 and previous research [25], it is thought that many-to-many mappings from movement parameters to musical parameters provide the most promising sound production.

As future research we suggest to use the findings from Chapter 5 as a basis for an investigation to usable musical mappings. Eventually this should result in a mature DMI, utilizing both gestural control types. We further suggest to perform a quantitative approach on the thesis that between and within participants there

exists a low amount of motional type variance. Also the HMM classifier should be made more robust for inter-subject variance. Moreover, the eventual goal for this system is to be made usable and accessible for others, sharing the freedom of making music in the air.

Bibliography

[1] http://musichackday.org/. Website, 2010. http://musichackday.org/.

[2] The international conference on new interfaces for musical expression. Website, 2011. http://www.nime.org/.

[3] Cycling 74. Max: Interactive visual programming environment for music, audio, and media. Website, 2010. http://cycling74.com/products/maxmspjitter/.

[4] NASA: National Aeronautics and Space Administration. Anthropometry and biomechanics. *Man-Systems and Integration Standards.*, 1, 2011.

[5] S. Arom. African polyphony and polyrhythm. *Cambridge, UK: Cambridge University Press.*, 1991.

[6] G. Becking. *De musikalische Rhythmus als Erkenntnisquelle.* 1928.

[7] G. Bertini and P. Carosi. Light baton: A system for conducting computer music performance. *In Proceedings of the International Computer Music Conference*, pages 73–76, 1992.

[8] F. Bevilacqua, B. Zamborlin, A. Sypniewski, and N. Rasamimanana. Continuous realtime gesture following and recognition. *Lecture Notes in Computer Science: Gesture in Embodied Communication and Human-Computer Interaction.*, pages 73–84, 2010.

[9] C. Cadoz and M. Wanderley. Gesture - music. *IRCAM*, 2000.

[10] J. Chadabe. The electronic century part i: Beginnings. *Electronic Musician*, pages 74–89, 2000.

[11] I. Cross. Music, cognition, culture and evolution. *The cognitive neuroscience of music.*, pages 42–56, 2003.

[12] Sofia Dahl. Playing the accent - comparing striking velocity and timing in an ostinato rhythm performed by four drummers. *Acta Acustica*, 90:762 – 776, 2004.

[13] H. Davies. Denis d'or. *Grove Music Online. Oxford Music Online.*, 2009.

[14] P. Dhawale, M. Masoodian, and B. Rogers. Barehand 3d gesture input to interactive systems. *New Zealand Chapters International Conference on Computer-Human Interaction: Design Centered HCI (CHINZ)*, pages 25–32, 2006.

[15] Richard O. Duda, Peter E. Hart, and David G. Stork. *Pattern Classification.* Wiley, New York, 2. edition, 2001.

[16] Z. Eitan and R.Y. Granot. How music moves: Musical parameters and listeners' images of motion. *Music Perception*, 23(3):221–247, 2006.

[17] Wilson et al. Marking system with pen-up/pen-down tracking. *United States Patent 5434370*, 1995.

[18] B Fry and C. Reas. Processing. Website, 2010. http://processing.org/.

[19] J. Oh G. Wang, N. Bryan and R. Hamilton. Stanford laptop orchestra (slork). *International Computer Music Conference*, 2009.

[20] A Glinsky. *Theremin: Ether Music and Espionage.* Illinois: University of Illinois Press., 2000.

[21] D. Goldberg. Unistrokes for computerized interpretation of handwriting. *United States Patent 5596656*, 1997.

[22] D Goldberg and C. Richardson. Touch-typing with a stylus. *Proceedings of INTERCHI93*, pages 80–87, 1993.

[23] F. Haflich and M. Burns. Following a conductor: The engineering of an input device. *In Proceedings of the International Computer Music Conference*, 1983.

[24] R.S. Hatten. Interpreting musical gestures, topics, and tropes, mozart, beethoven, schubert. *Bloomington: Indiana University Press.*, 2004.

[25] A. Hunt and R. Kirk. Mapping strategies for musical performance. *Trends in Gestural Control of Music.*, 2000.

[26] Native Instruments. Absynth 5. Website, 2011. http://www.native-instruments.com/#/en/products/producer/absynth-5/.

[27] P. Isokoski and R. Raisamo. Device independent text input: A rationale and an example. *Advanced Visual Interfaces 2000*, pages 76–83, 2000.

[28] A.R. Jensenius, M.M. Wanderley, R.I. Godoy, and M Leman. *Musical Gestures: Sound, Movement and Meaning.* Routledge, 2010.

[29] D Keane and P Gross. The midi baton. *In Proceedings of the International Computer Music Conference.*, pages 151–154, 1989.

[30] C. Kenner. Glovepie. Website, 2011. http://sites.google.com/site/carlkenner/glovepie.

[31] N Krishnamurthy, D Bacher, J.R. McFerron, and A.P. Batista. Wiimocap: A low-cost motion capture system using the nintendo wiimote. *Society for Neuroscience Abstracts*, 2009.

[32] J.C. Lee. Hacking the nintendo wii remote. *IEEE Pervasive Computing*, 7:39–45, 2008.

[33] J.C. Lee. Tracking your fingers with the wiimote. Website, 2010. http://johnnylee.net/projects/wii/.

[34] S.U. Lee and I. Cohen. 3d hand reconstruction from a monocular view. pages III: 310–313, 2004.

[35] M. Leman. *Embodied Music Cognition and Mediation Technology.* The MIT Press, 2007.

[36] J.S. Lipscomba. A trainable gesture recognizer. *Pattern Recognition*, 24:895–907, 1990.

[37] G. Luck and Nte. Sol. An investigation of conductors' temporal gestures and conductor. *Psychology of Music*, 36:81–99, 2008.

[38] T. Marrin and J. Paradiso. The digital baton: A versatile performance instrument. *In Proceedings of the International Computer Music Conference*, pages 313–316, 1997.

[39] M. Mathews and F. Moore. Groovea program to compose, store and edit functions of time. *Communications of the ACM*, 1970.

[40] M.V. Mathews. The radio baton and the conductor program. *Computer Music Journal*, (15):37–46, 1991.

[41] D. McNeill. Hand and mind: What gestures reveal about thought. *University of Chicago Press*, 1992.

[42] D. McNeill. Language and gesture. *Cambridge: Cambridge University Press.*, 2000.

[43] D. McNeill. Gesture and thought. *University of Chicago Press*, 2005.

[44] R. Middleton. Popular music analysis and musicology: bridging the gap. *Popular Music*, 12:177–190, 2001.

[45] E.R. Miranda and M.M. Wanderley. *New Digital Musical Instruments: Control And Interaction Beyond The Keyboard.* A-R Editions, 2006.

[46] N.H. Rasamimanana and F. Bevilacqua. Effort-based analysis of bowing movements: evidence of anticipation effects. *The Journal of New Music Research*, 37:339–351, 2009.

[47] N.H. Rasamimanana and F. Bevilacqua. Perspectives on gesture-sound relationships informed from acoustic instrument studies. *Organised Sound*, 14:208–216, 2009.

[48] B.H. Repp. Musical motion: Some historical and contemporary perspectives. *Royal Swedish Academy of Music*, SMAC 93(79):128–135, 1993.

[49] T. Schlomer, B. Poppinga, N. Henze, and S. Boll. Gesture recognition with a wii controller. *University of Oldenburg*, 2010.

[50] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake. Real-time human pose recognition in parts from single depth images. *Computer Vision and Pattern Recognition*, 2, 2011.

[51] P. Toiviainen and G. Luck. Embodied meter: Hierarchical eigenmodes in music-induced movement. *Music Perception*, pages 59–70, 2010.

[52] T Troillard and W. Troillard. Osculator. Website, 2010. http://www.osculator.net/.

[53] D. Trueman and P. Cook. Plork : The princeton laptop orchestra. Website, 2010. http://plork.cs.princeton.edu/.

[54] A. Truslit. *Gestaltung und Bewegung in der Musik.* 1938.

[55] D. Venolia and F. Neiberg. T-cube: Self-discrolsin pen-based alphabet. *Proceedings of the CHI 94*, pages 265–270, 1994.

[56] N.L. Wallin, B. Merker, and S. Brown. The origins of music. 2000.

[57] Y. Wang and J. Popovic. Real-time hand-tracking with a color glove. *Proceedings of ACM SIGGRAPH*, 28, 2009.

[58] G. Welch and E.; Foxlin. Motion tracking: no silver bullet, but a respectable arsenal. *Computer Graphics and Applications*, 22:24–38, 2002.

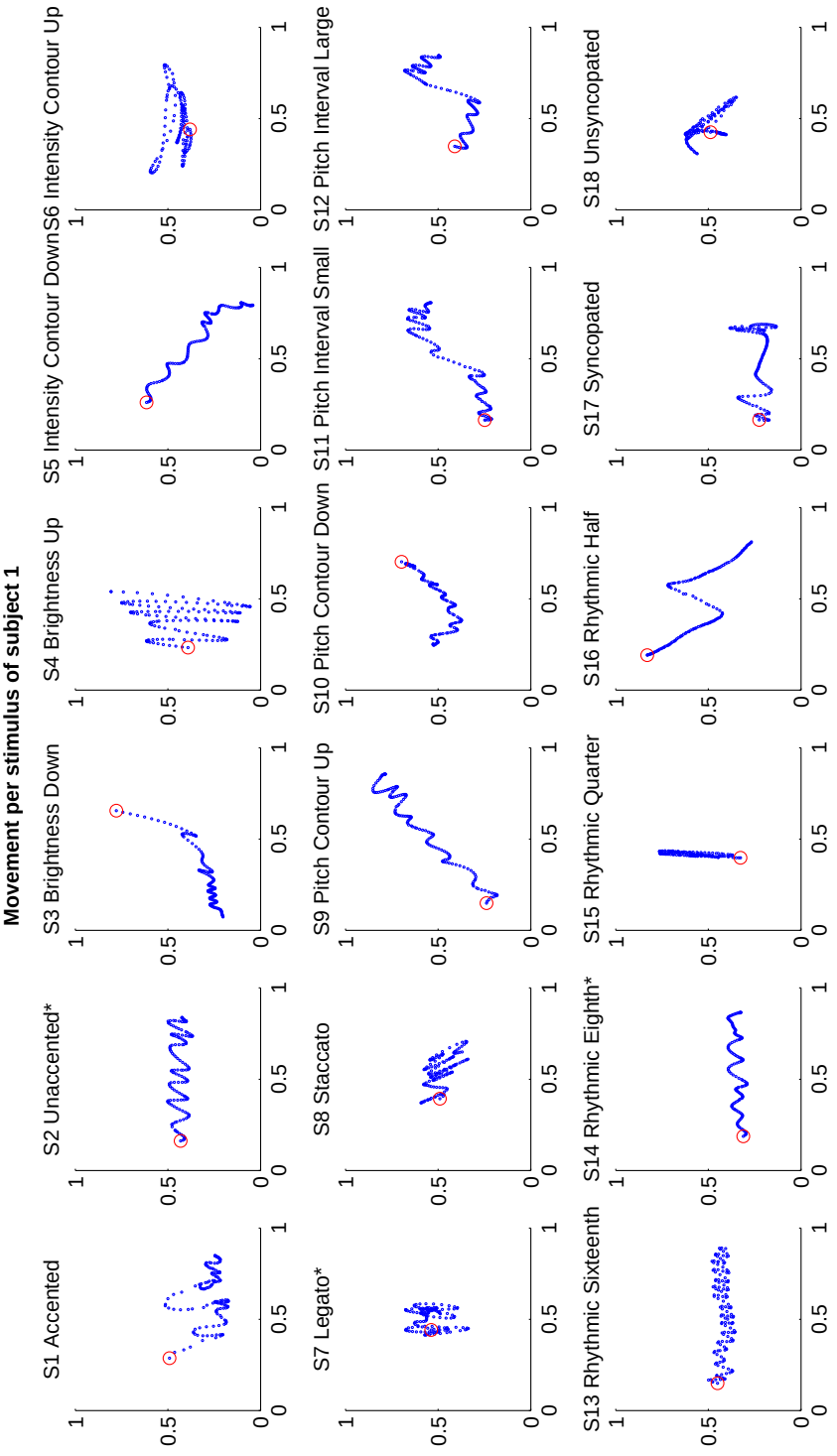
[59] D. Willems, R. Niels, M. van Gerven, and L. Vuurpijl. Iconic and multi-stroke gesture recognition. *Pattern Recognition*, 42(12):3303–3312, 2009.

[60] Y.S. Xu, J.H. Gu, Z. Tao, and D. Wu. Bare hand gesture recognition with a single color camera. pages 1–4, 2009.

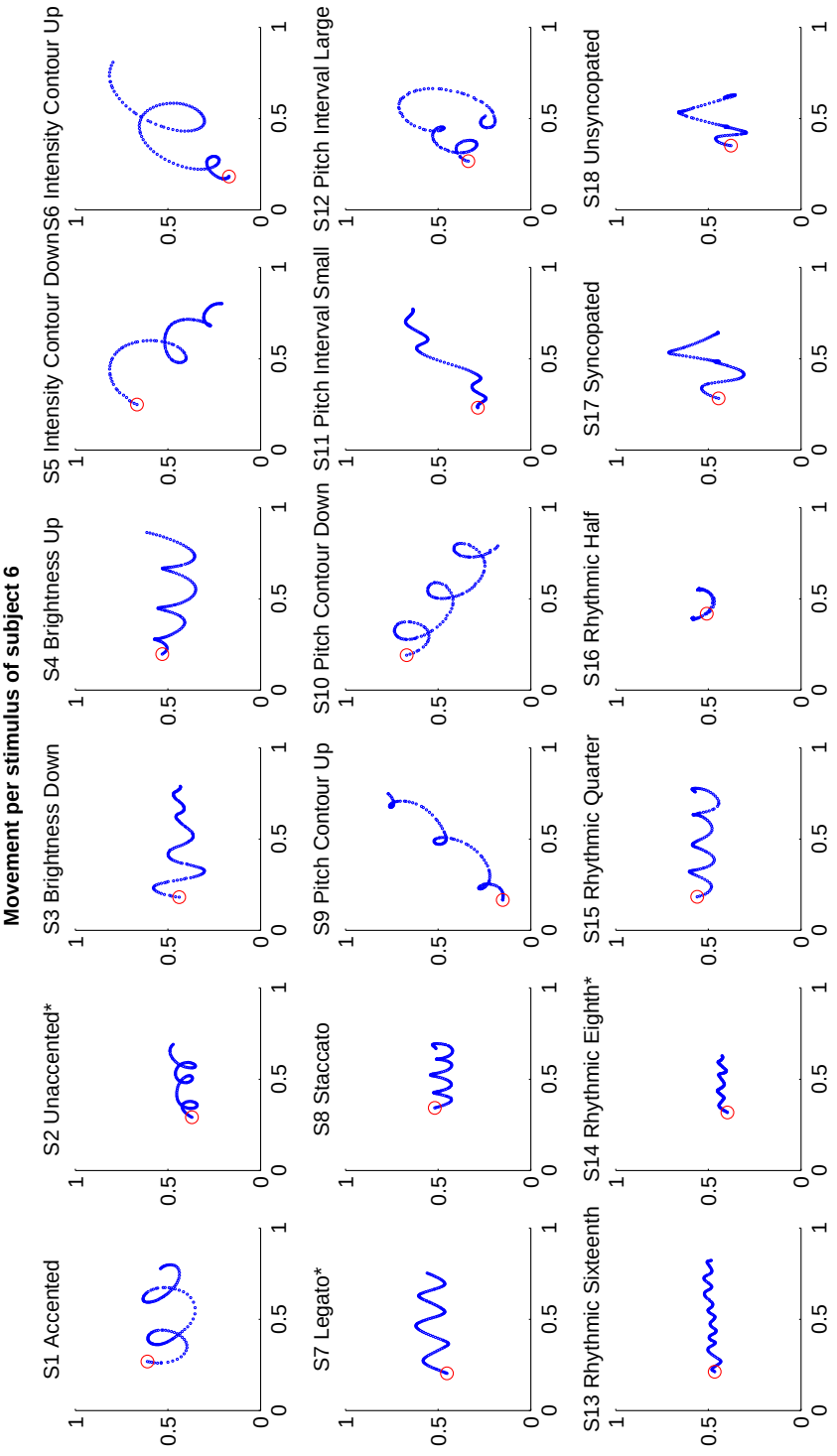
[61] L. Zhao. Synthesis and aquisition of laban movement analysis qualatative parameters for communicative gestures. *PhD thesis*, 2001.

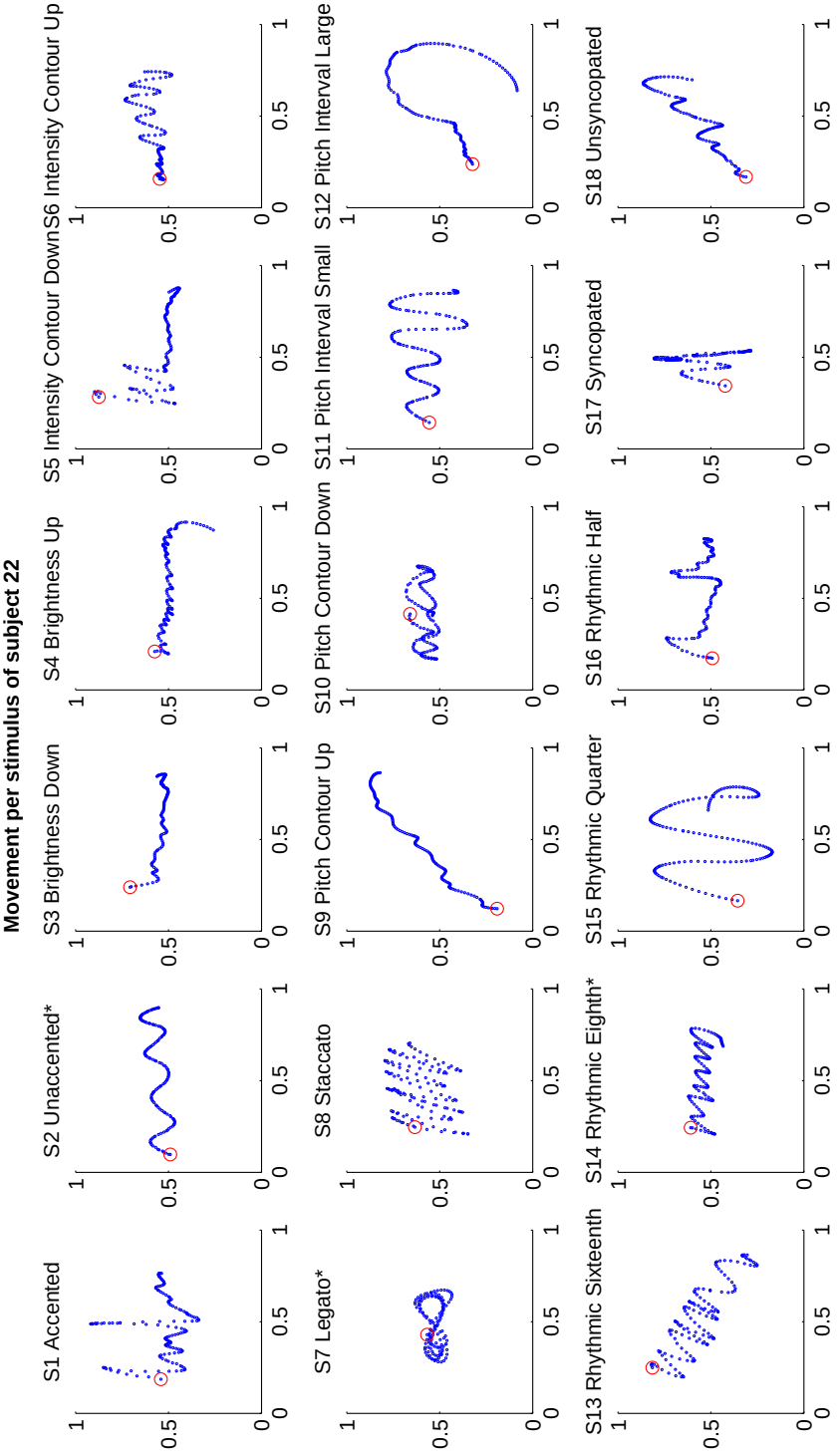
Appendix A

Examples of Participants' Movements


Movement per stimulus of subject 1
S1 Accented
S2 Unaccented*
S3 Brightness Down
S4 Brightness Up
S5 Intensity Contour Down
S6 Intensity Contour Up
S7 Legato*
S8 Staccato
S9 Pitch Contour Up
S10 Pitch Contour Down
S11 Pitch Interval Small
S12 Pitch Interval Large
S13 Rhythmic Sixteenth
S14 Rhythmic Eighth*
S15 Rhythmic Quarter
S16 Rhythmic Half
S17 Syncopated
S18 Unsyncopated

Movement per stimulus of subject 2
S1 Accented
S2 Unaccented*
S3 Brightness Down
S4 Brightness Up
S5 Intensity Contour Down
S6 Intensity Contour Up
S7 Legato*
S8 Staccato
S9 Pitch Contour Up
S10 Pitch Contour Down
S11 Pitch Interval Small
S12 Pitch Interval Large
S13 Rhythmic Sixteenth
S14 Rhythmic Eighth*
S15 Rhythmic Quarter
S16 Rhythmic Half
S17 Syncopated
S18 Unsyncopated


Movement per stimulus of subject 6
S1 Accented
S2 Unaccented*
S3 Brightness Down
S4 Brightness Up
S5 Intensity Contour Down
S6 Intensity Contour Up
S7 Legato*
S8 Staccato
S9 Pitch Contour Up
S10 Pitch Contour Down
S11 Pitch Interval Small
S12 Pitch Interval Large
S13 Rhythmic Sixteenth
S14 Rhythmic Eighth*
S15 Rhythmic Quarter
S16 Rhythmic Half
S17 Syncopated
S18 Unsyncopated


Movement per stimulus of subject 22
S1 Accented
S2 Unaccented*
S3 Brightness Down
S4 Brightness Up
S5 Intensity Contour Down
S6 Intensity Contour Up
S7 Legato*
S8 Staccato
S9 Pitch Contour Up
S10 Pitch Contour Down
S11 Pitch Interval Small
S12 Pitch Interval Large
S13 Rhythmic Sixteenth
S14 Rhythmic Eighth*
S15 Rhythmic Quarter
S16 Rhythmic Half
S17 Syncopated
S18 Unsyncopated