Proceedings of the International Conference on New Interfaces for Musical Expression, 30 May - 1 June 2011, Oslo, Norway

SoundSaber - A Motion Capture Instrument

Kristian Nymoen, Ståle A. Skogstad
fourMs group - Music, Mind, Motion, Machines
Department of Informatics
University of Oslo, Norway
{krisny, savskogs}@ifi.uio.no

Alexander Refsum Jensenius
fourMs group - Music, Mind, Motion, Machines
Department of Musicology
University of Oslo, Norway
a.r.jensenius@imv.uio.no

ABSTRACT
The paper presents the SoundSaber - a musical instrument based on motion capture technology. We present technical details of the instrument and discuss the design development process. The SoundSaber may be used as an example of how high-fidelity motion capture equipment can be used for prototyping musical instruments, and we illustrate this with an example of a low-cost implementation of our motion capture instrument.

1. INTRODUCTION
We introduce the SoundSaber, a musical instrument based on optical infrared marker-based motion capture technology. *Motion capture* (mocap) involves recording motion, and translating it to the digital domain [10]. *Optical* motion capture means that the system is based on video cameras, and we distinguish between *marker-based* and *markerless* systems which work without markers. We will refer to musical instruments based on optical motion capture as *mocap instruments*.

Optical infrared marker-based mocap technology is superior to most other methods of motion capture with respect to temporal and spatial resolution. Some systems can track markers at a rate of more than 1000 frames per second, and in most cases they provide a spatial resolution in the sub-millimeter range. On the other hand, this technology is expensive, and better suited for laboratory use than for stage performances. A wide range of other less expensive and portable mocap technologies exists, like accelerometer-based sensor systems and computer vision. These provide different types of data, usually with lower frame rate and spatial resolution than optical infrared mocap.

A large amount of the research that is done in our lab involves the exploration of motion capture systems for musical interaction, ranging from high-end technologies to solutions like web-cameras and accelerometers. This involves studies of the different technologies separately, and also experiments on how the experience from interactive systems based on high-end mocap technology can be transferred to low-cost mocap technologies.

We present the SoundSaber as an example of how a seemingly simple sound synthesiser may become interesting through the use of high quality motion capture technology and an intuitive action-sound model. With a system that is able to register very subtle motion at a high sampling rate, it is possible to create an instrument that comes close to the control intimacy of acoustic instruments [11]. These ideas are presented through reflections that have been made while developing the instrument. Included in the presentation are some thoughts and experiences from how optical motion capture technology can be used to prototype new interfaces for musical expression.

In Section 2 we lay out a general theory on digital musical instruments and use of mocap for sound generation. Section 3 presents the SoundSaber, including considerations and evaluations that have been made in the process of development. In Section 4 we illustrate how the instrument was "ported" to another technology and compare the results to the original SoundSaber. Section 5 provides conclusions and directions for future work.

2. MOCAP INSTRUMENT CONTROLLERS
Most digital musical instruments consist of a controller with sensors, a sound synthesiser, and a defined *mapping* between the control data from the sensors and the input parameters of the synthesiser [5]. Mocap instruments are slightly different in that the controller is separate from the sensor technology. This distinction between the sensors and the controller present an interesting opportunity because almost any object can be used to communicate with the mocap system: a rod, a hand, an acoustic instrument, etc.

This makes it possible to try out objects with different physical properties and shapes, hence also different *affordances*. In design literature, the affordance of an object is a term used to describe the perceived properties of how this object could possibly be used [6]. For an object used in a mocap instrument, the affordance may refer to a "pool" of different control actions that could be associated with it, e.g. whether it should be held with one or both hands. Following this, physical properties of the object, such as size, inertia, etc., will also influence how it can be handled. The possibility of quickly swapping objects may be a useful tool for prototyping new digital musical instruments.

The data from the motion capture system can be processed in several ways, see [1] and [10] for discussion on how motion capture data can be mapped to musical parameters. The GrainStick installation at IRCAM used mocap technology to generate sound in yet another way, using the metaphor of a virtual rainstick being held between two objects [4]. Our choices for data processing in the SoundSaber will be presented in Sections 3.2 to 3.5.

3. THE SOUNDSABER
The different components of the SoundSaber are illustrated in Figure 1. The position of the controller is captured by the motion capture system, which sends position data to a

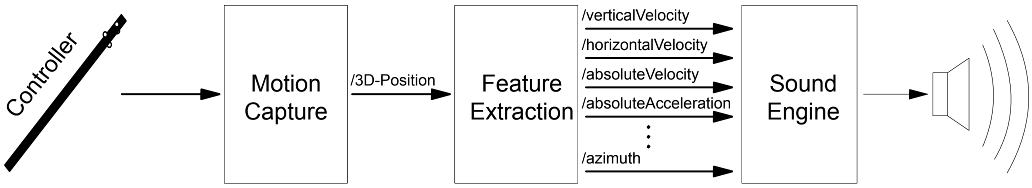

Figure 1: The figure shows the different parts of the SoundSaber instrument from controller, via motion capture technology and feature extraction to a sound synthesiser.

Max/MSP patch that calculates various features from the data. The features calculated by this patch are mapped to various control parameters in a synthesiser.

One of the advantages of digital musical instruments is that it is simple to try out different technologies for each part of the instrument. In our own work, we have experimented with different motion capture systems and controllers. Currently, we have two versions of the SoundSaber: The original one, based on optical motion capture, and another wii-controller (wiimote) implementation.

We will start the presentation of the SoundSaber by describing the controller, followed by a presentation of the motion capture technology, feature extraction and the synthesiser. Even though the different parts of the instrument are presented separately, they have been developed together, both simultaneously and iteratively.[1]

3.1 The controller

The SoundSaber controller that we are currently using is a rod, roughly 120 cm in length with a diameter of 4 cm, and is shown in Figure 2. Four markers are placed in one end of the rod, and the motion capture system recognizes these as a single *rigid object*, tracking position and orientation of the tip of the rod. The rod is heavy enough to give it a reasonable amount of inertia, and at the same time light enough so that it does not feel too heavy, at least not when it is held with both hands. The shape and mass of the rod also make it natural to perform large and smooth actions. We have observed that the majority of people who have tried the instrument performed gestures that imitate fencing. The reason for this may be their association of these gestures with the name of the instrument in combination with the physical properties and affordance of the controller.

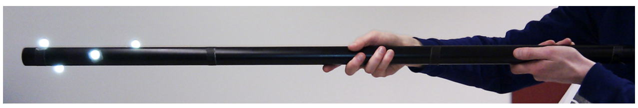

Figure 2: The SoundSaber controller

3.2 Motion capture

We have been using different motion capture systems for the SoundSaber. Initially we used an 8-camera OptiTrack system from NaturalPoint, which can stream real-time data at a rate of 100 Hz. The OptiTrack software uses the proprietary NatNet protocol for data streaming. We used a client developed by Nuno Diniz at IPEM in Ghent for translating NatNet data to Open Sound Control (OSC) over UDP. OSC simplifies the communication between the motion capture system and the layers for feature extraction, mapping and sound synthesis.

More recently, we have been using a high-end motion capture system from Qualisys. This system has a higher spatial resolution than OptiTrack, and it is able to stream data at higher sampling rates. The Qualisys system also has native support for Open Sound Control.

3.3 Feature extraction

We have implemented a tool in Max/MSP for real-time feature extraction from position data. Our approach is similar to the Motion Capture Music toolbox, developed by Dobrian et al. [1], with some differences. Our tool is structured as one single module, and outputs data as OSC messages. OSC formatting of these features simplifies the mapping between the motion features and the control features in the synthesiser.

Thus far, difference calculations, dimensionality reduction and transformations between different coordinate systems have been implemented. Based on a three-dimensional position stream the patch calculates:

- Velocity in a single direction, e.g. vertical velocity
- Velocity in a two-dimensional subspace, e.g. horizontal velocity
- Absolute velocity, as the vector magnitude of the three velocity components
- Change in absolute velocity
- Acceleration in a single direction
- Absolute acceleration
- Polar equivalent of the cartesian input coordinates, providing horizontal angle, elevation, and distance from the origin

3.4 Sound synthesis

As the name *SoundSaber* suggests, we initially had an idea of imitating the sound of the lightsaber from the Star Wars movies. The development of the synthesiser was more or less a process of trial and error to find a sound that would have some of the perceptual qualities that are found in the lightsaber sound.

The SoundSaber synthesiser is implemented in Max/MSP. Figure 3 shows a schematic illustration of the synthesiser, where a pulse train (a sequence of impulses or clicks) with a frequency of 1000 Hz is sent through two delay lines with feedback loops. The delay times for the delay lines can be adjusted by the user, resulting in variations in harmonic content. Furthermore, the output from the delay lines is sent to a ring modulator where it is modulated by a sinusoidal oscillator. The user can control the frequency of this oscillator in the range between 40 and 100 Hz. The ring modulated signal and the output from the delay lines are added together and sent through an amplitude control, then another feedback delay line and finally through a bandpass filter where the user controls bandwidth and frequency.

3.5 Mapping

Several considerations have been made regarding the action-sound relationship in the SoundSaber. Naturally, we have

[1]For video examples of the SoundSaber, please visit `http://www.youtube.com/fourmslab`

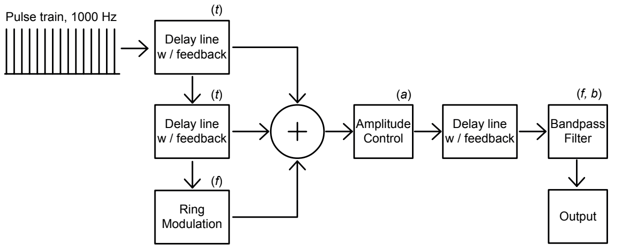

Figure 3: The SoundSaber synthesiser. Letters in parentheses denote user-controllable parameters: (t) = time, (f) = frequency, (a) = amplitude and (b) = bandwidth.

not been limited to mimicking action-sound relationships from traditional musical instruments, but at the same time we do appreciate some of the constraints that acoustic instruments provide. For instance, the sound of an instrument is almost always the result of an energy transfer from a sound-producing action to mechanical vibrations.

Since our approach to the physical design of this instrument has been simple, using only the position of a single point on the controller as the basis for feature extraction, we have chosen a simple approach when mapping motion features to control parameters. This is what Hunt and Wanderley call *explicit mapping*, meaning direct couplings between motion features and control parameters in the sound synthesiser [2].

When designing mapping for a mocap instrument, it is important to understand what the motion features actually describe. Motion features calculated from a stream of data describing position of a controller in a room can be one (or a combination) of the following:

Relative to the room meaning that the axes of the room influence the motion feature. An example of this is the vertical velocity component of the motion.

Relative to the controller itself typically referring to difference calculations, e.g. the absolute velocity.

Relative to another controller describing how the controller relates to other controllers in the room. For instance the distance to another SoundSaber.

In the current SoundSaber implementation, we have only used data that describes the controller in relation to the room or to itself. But we believe that the perspective of how the instrument relates to other controllers presents interesting possibilities in making collaborative musical instruments.

One of the considerations we have made is regarding motion in the horizontal plane. Should it make a difference whether the instrument is being moved along the X-axis or the Y-axis? In our opinion, the SoundSaber should respond equally whether the musician is on the left side or the right side of a stage, and also behave in the same manner no matter which direction the performer is facing, as is the case for any hand-held acoustic instrument. Therefore we reduced the two dimensions of horizontal velocity to a single absolute horizontal velocity, and let this mapping govern one of the timbral control parameters in the synthesiser (the delay time of the first delay line).

Vertical motion, on the other hand, is different. Our previous experiments have shown that people tend to relate vertical motion to changes in frequency, such as changes in pitch and spectral centroid [7, 8]. No matter which direction the performer is facing, gravity will act as a natural reference. In light of this, we have chosen to let the vertical position control the frequency of the ring modulation and the bandpass filter, and the vertical velocity control the delay time of the second delay line.

Another action-sound relationship which has been confirmed in our previous experiments, is the correspondence between velocity and loudness [7]. Hunt and Wanderley noted that increased input energy is required to increase sound energy in acoustic instruments, and received better results for a digital musical instrument where users had to feed the system with energy to generate sound, rather than just positioning a slider to adjust sound level [2]. With this in mind, we wanted an increase in kinetic energy to result in an increase in sound energy. Therefore, we let the absolute velocity control the amplitude of the synthesiser.

We have implemented a simple mapping for sound spatialisation. Spatial sound is obviously related to the room, so we used motion features related to the room in this mapping. More specifically, we sent the polar position coordinates of the rod to a VBAP control system [9], so the musician can control sound spatialisation by pointing the SoundSaber towards different loudspeakers.

3.6 SoundSaber evaluation

Neither the feature extraction, the explicit mapping strategy, nor the synthesiser of the SoundSaber are particularly sophisticated or novel by themselves. At the same time, after observing how people interact with the instrument, we feel confident to say that such interaction is engaging for the user. We believe that the most important reason for this are the considerations that were made to obtain a solid coupling between control actions and sound.

In addition to the rod, we tried using three other objects for controlling the SoundSaber synthesiser. For two of these, we simply changed the rod with another object and used the same motion capture technology, meaning that the only difference was the object itself. First, we tried a small rod, which was best suited for single-hand use, and also had less inertia and thus higher mobility. Second, we tried using a small handle with markers. This handle reduced the distinction between the controller and the performer, because the motion of the controller was basically equal to the hand motion of the performer. Both of these solutions were less satisfying than the large rod because the loudness control in the synthesiser had a fairly long response time, making it more suitable for controllers with more inertia. Also, the deep and full sound of the SoundSaber works better with a larger object. Third, as mentioned above, we made an implementation of the SoundSaber using a Nintendo Wii controller which will be discussed in more detail below.

Furthermore, we believe that the considerations of how motion features related to sound were important. The use of vertical position (which is only relative to the room) to adjust spectral centroid via a bandpass filter, and of absolute velocity (which is only relative to the object itself) to control loudness appeared to work well.

Nevertheless, the complexity of the control input, and the motion capture system's ability to capture motion nuances are perhaps the most important reasons why it is engaging to interact with the SoundSaber. Even though separate motion features were selected and mapped to different control parameters, the motion features themselves are related to each other. As an example, consider what happens when the performer makes a change in vertical position of the rod to adjust the spectral centroid. This action will also imply a change in the motion features "vertical velocity" and "absolute velocity".

When the spatial and temporal resolution of the motion capture system is high, the instrument responds to even the smallest details of the performer's motion. For a reasonably sized object like the SoundSaber rod, we are satisfied with a spatial resolution of 1 mm and a frame rate of 100 Hz, but for smaller and more responsive objects we might require even higher resolution to capture the nuances of the actions these objects afford.

4. TOWARDS PORTABILITY

Because of the expensive hardware, the implementation of the SoundSaber based on optical motion capture is not available to everyone. One motivation for this research is to make instruments that are based on high-end technology available to a broader audience. Thus, we need less expensive and preferably also more portable solutions.

Of the many affordable sensor solutions, we chose to use a Nintendo wii-controller (wiimote) for our low-cost implementation. The wiimote provides a different set of control possibilities than optical motion capture, and the major challenges with porting the SoundSaber to the wiimote are related to processing the data from the controller and mapping strategies. A survey by Kiefer et al. ([3]) showed that the wiimote could be well suited for continuous control, which makes it an interesting test case for the SoundSaber.

4.1 Wiimote implementation

We used OSCulator[2] for communication between the wiimote and the computer. OSCulator provides estimates of orientation and absolute acceleration of the wiimote.

Orientation data can be seen as similar to the position data from the motion capture system, in the sense that it describes a state of the device within a single time-frame. Because of this similarity, change in orientation was mapped to the amplitude control. Although ideally the orientation data from the wiimote should not change unless there was an actual change in the orientation of the wiimote, the fact is that these values changed quite a lot even for non-rotational motion. Because of a significant amount of noise in the data, we used one of the push-buttons on the wiimote as an on/off button, to prevent the instrument from producing sound when the controller was lying still.

The angle between the floor and an imagined line along the length axis of the wiimote is called *pitch*. We let this value and its derivative control the synthesis parameters that originally were controlled by vertical position and vertical velocity, meaning the first delay line, frequency of the bandpass filter and the frequency of the ring modulator. Finally, we let the estimate of the dynamic acceleration control the second delay line in the synthesis patch.

4.2 Evaluation of the wiimote implementation

The wiimote implementation of the SoundSaber was, as expected, not as satisfying as the version based on optical motion capture. In our experience the orientation values needed some time to "settle". By this we mean that sudden actions affected these parameters quite a lot, and they did not settle at stable values until after the wiimote stopped moving. As a result, an action that was meant to cause a sudden increase in frequency would cause a sudden increase in loudness when the action started, and then a sudden increase in frequency when the wiimote was being held steady pointing up.

Using the tilt parameter *pitch* with the wiimote is conceptually quite different from the original mapping, where vertical position was used. However, we were surprised by how well this worked for slower motion. During a demonstration, one subject was moving the wiimote up and down with his arm fully stretched out, not realising that by doing this, he also pointed the wiimote up and down. The subject was puzzled by this and asked how we were able to extract vertical position values from the accelerometer in the wiimote.

In our opinion, the most important differences between the high-end implementation and the wiimote version are the size of the controller and the accuracy of the data. The wiimote data is too noisy for accurate control, and the size and shape of the wiimote afford one-handed, rapid impulsive actions, in contrast to the rod which is more suited for larger and slower actions. The wiimote implementation would probably benefit from using another synthesis module that is better suited for its affordances.

[2] `http://www.osculator.net/`

5. CONCLUSIONS AND FUTURE WORK

In this paper we presented the SoundSaber and our thoughts on how optical motion capture technology can be used for prototyping musical instruments. Our experience shows us that even a quite simple synthesiser and simple control signal are sufficient to create an interesting musical instrument, as long as the action-sound coupling is perceptually robust.

We will continue our work on the SoundSaber and other mocap instruments. It would be interesting to investigate whether the instrument would benefit from attaching an FSR. Furthermore, we see intriguing challenges and research questions related to developing the SoundSaber into a collaborative instrument, as well as an adaptive instrument that will adjust to different performers and situations.

6. REFERENCES

[1] C. Dobrian and F. Bevilacqua. Gestural control of music: using the vicon 8 motion capture system. In *Proceedings of NIME 2003*, pages 161–163, Montreal, Canada, 2003.

[2] A. Hunt and M. M. Wanderley. Mapping performer parameters to synthesis engines. *Organised Sound*, 7(2):97–108, 2002.

[3] C. Kiefer, N. Collins, and G. Fitzpatrick. Evaluating the wiimote as a musical controller. In *Proceedings of ICMC 2008*, Belfast, Northern Ireland, 2008.

[4] G. Leslie et al. Grainstick: A collaborative, interactive sound installation. In *Proceedings of ICMC 2010*, New York, USA, 2010.

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

[6] D. A. Norman. *The Design of Everyday Things*. Basic Books, New York, 1988.

[7] K. Nymoen. Comparing sound tracings performed to sounds with different sound envelopes. In *Proceedings of FRSM - CMMR 2011*, pages 225 – 229, Bhubaneswar, India, 2011.

[8] K. Nymoen, K. Glette, S. A. Skogstad, J. Torresen, and A. R. Jensenius. Searching for cross-individual relationships between sound and movement features using an SVM classifier. In *Proceedings of NIME 2010*, pages 259 – 262, Sydney, Australia, 2010.

[9] V. Pulkki. Generic panning tools for max/msp. In *Proceedings of ICMC 2000*, pages 304–307, 2000.

[10] S. A. Skogstad, A. R. Jensenius, and K. Nymoen. Using IR optical marker based motion capture for exploring musical interaction. In *Proceedings of NIME 2010*, pages 407–410, Sydney, Australia, 2010.

[11] D. Wessel and M. Wright. Problems and prospects for intimate musical control of computers. *Computer Music Journal*, 26:11–22, September 2002.