Master's Project
HCC Project Seminar 2011, 15 ECTS
Supervisor: Esben Warming Pedersen

Sputnik

Simon Wallner*

22nd December 2011

This paper evaluates *Sputnik* a 3D environment with which the user can freely interact through an elastic *arc of light/fishing rod* metaphor, to explore, create and interact with virtual *sound objects*. These sound objects are placed in the scene and react to the user's input by sending MIDI commands to an external audio program thus creating or manipulating the sound.

1. Reading Guide

This report is aimed at readers with a general knowledge in computer science on the bachelor's level and above. Prior knowledge and experience with HCC and Tangible User Interfaces (TUI) is recommended, but the related work section should be a good starting point for interested readers new to this field.

This report is accompanied by a short video that gives a short overview and demonstrates the main interactions with Sputnik. It is advised to watch this video before reading the report.

Also accompanying this report are the questionnaire and evaluation form used for the user study. They can be found at the root of this package named `questionnaire.pdf` and `evaluation-form.pdf` and in appendix A and B.

*`me@simonwallner.at`

2. Introduction

Computer music has been around us for some time now and through the use of the computer, musicians have sheer endless possibilities of musical expression. With this plethora of possibilities comes the need for constraints and control to harness this expressive potential. Over the recent years many standard and non-standard interface have been developed, ranging from the ordinary button-fader-nob MIDI interface to more elaborate interfaces and systems like the *reactable*[Jordà et al., 2007], *mixiTUI*[Pedersen and Hornbæ k, 2009], or commercial solutions like the *Novation Launchpad*[1] or *Native Instruments Maschine*[2] to name but a few.

With the advent of motion based controllers in consumer entertainment systems, marked by the release of the *Wii*[3] console in late 2006, motion controllers became widely and cheaply available. This and their interface capabilities make them the ideal tools to explore the realm of NIMEs.

A common problem of computer music interfaces is that often the process of sound creation is not readily comprehensible. Seeing a performer on stage behind their laptop twisting knobs and adjusting faders might be ambiguous to an uninformed observer. It can be hard to relate the artist's action to the resulting sounds. This can hinder the experience and might go as far as to the point where the audience suspects that an artist just pressed play, as interviews conducted by Pedersen and Hornbæ k [2009] show.

This paper introduces *Sputnik*, a system that uses a *Wiimote* controller to interact with a dynamic 3D scene. In this scene, a variety of sound creating objects are placed that send MIDI signals to an external audio program upon the user's interaction.

Users can freely navigate the 3D scene and interact with it through an elastic *arc of light/fishing rod* metaphor. It seems as if the *arc of light* was coming out of the Wiimote and reaches into the scene, acting as an extension of the user's body into the virtual space. With this bodily extension, users can *grab* and *drag* objects around the 3D scene.

Sputnik is evaluated in s small user study and the findings and future work directions are discussed. As part of this, the *arc or light* metaphor is evaluated as a way to spatially manipulate objects in 3D space. Parallels to tangible user interfaces (TUI)[Ishii and Ullmer, 1997] and specifically spatial TUIs[Sharlin et al., 2004] are drawn.

[1] http://www.novationmusic.com/products/midi_controllers/launchpad
[2] http://www.native-instruments.com/#/en/products/producer/maschine/
[3] http://de.wikipedia.org/wiki/Wii

3. Paper Outline

The following section gives an overview over related work in the field of NIMEs and tangible user interfaces. Section 5 goes into detail about Sputnik, both on a conceptual and a technical level. Section 9 describes the performed user study and the paper is finally concluded in section 10 where the findings are discussed.

4. Related Work

Sputnik draws from a wide variety of related work and this section is therefore loosely split into more practical and more theoretical work.

Practical Work Only a few projects exist that go into a similar direction as Sputnik. The *Virtual Xylophone*[Mäki-Patola et al., 2005] is a virtual reality system in which the user can place xylophone bars of different pitch in the scene and then strike them with a virtual mallet. By translating the configuration and mapping of the real instrument into the VR environment, new modes of play emerge as it is shown in the paper. Zappi et al. [2010] created a virtual controller for *Ableton Live*[4] that allows users to create simple proxy objects in a VR environment, bind them to certain controls and use them effectively as virtual sliders. Rodet et al. [2005] created a virtual environment for an exhibition setting. Users interact with the system via a 6-DOF motion tracker with tactile feedback. However, the user's actions in the system are highly constrained.

More projects can be found in the realm of Tangible User Interfaces. An early project is *Audiopad*[Patten et al., 2002] that uses pucks on a table top interface to select and interact with sounds. With *mixiTUI*, Pedersen and Hornbæ k [2009] created a table top tangible interface for a sequencer that aimed not only to be functional but also to visually enrich the artist's performance. Interviews with musicians and an extensive user study have been performed. Jordà et al. [2007] created the famous *reacTable*, also a table top tangible interface that allows the creation and manipulation of music by composing various objects on its surface.

After the release of the Wii in late 2006, the Wiimote motion controller received some attention in and outside the field of musical interfaces: Kiefer et al. [2008] assess the general qualities of the Wiimote as a musical controller and Miller [2010] uses the Wiimote and sensor bar to create the *Wiiolin*, a virtual violin that mimics the real instrument and can be played either in an upright position like a cello or horizontally like a violin. It senses the button presses and tracks the movement of the *bow*, i.e., the sensor bar to create the sounds.

Similarly, Miyama [2010] uses an low resolution distance sensor array to control the many parameters of a synthesizer. A small gui application is merely used for monitoring the system's state, and sound creation is done in pd.

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

Theoretical Work The field of *Tangible User Interfaces (TUI)* provides part of the theoretical background for this work. Work of Fitzmaurice et al. [1995] introduced the term *graspable interface* and then later Ishii and Ullmer [1997] introduced the term and wider concept of TUI. Shaer [2009] gives a very good overview over this field as well as the history of TUI studies. Ullmer and Ishii [2000] introduced *MCRpd*, a formal model for describing and analysing TUIs.

Sharlin et al. [2004] introduced *spatial TUIs* that focuses on *I/O unification* by tightly coupling the action and perception spaces and embodying a clear state representation across all sensory modalities.

Entering the musical realm, Fels and Lyons [2011] give a good overview and general introduction into the field of NIMEs. Cook [2001] shares 13 general principles for designing computer music controllers that resulted from his long lasting experience in this field. Dobrian and Koppelman [2006] asks the question of virtuosity and expression and points out the lack of a standard repertoire.

In contrast, Gurevich and Treviño [2007] question the hegemonic composer–interpret–listener relation in favour of a more holistic *ecological* view of musical expression. Later work by Gurevich et al. [2010] evaluated a highly constrained, prototypical one-button instrument that spurred a wide variety of play styles in test users.

Closing the loop to design and HCI, Magnusson [2010] gives a good overview over the field of affordance and elaborates on contraints from different viewing angles and how they impact and support creativity. Finally, Wanderley and Orio [2002] go into depth over evaluating input devices for musical expression in the context of HCI.

5. Sputnik

Sputnik is a New Interface for Musical Expression (NIME) that combines 3D graphics with the capabilities of the wireless *Wiimote* and *Nunchuck* controller. The user is presented with a colourful 3D scene that contains various interactive objects. The user can freely navigate the scene and interact with these object to create sounds.

Sections 5.1 and 5.2 go into detail about the general set up and the virtual scene. Sections 5.3 and 5.4 explain the navigation and interaction, and finally section 5.5 describes the physics used in Sputnik.

5.1. Set Up

The system is set up in a room with an overhead mounted video projector. The IR-sensor bar, needed for the Wiimote controller, can either be placed on the upper or lower edge of the projected image. It consists of two IR emitters that allow the IR camera in the Wiimote to track its relative orientation in space. A Wiimote and Nunchuck controller are used and only a single person at a time can use the system.

Figure 1 illustrates the setup.

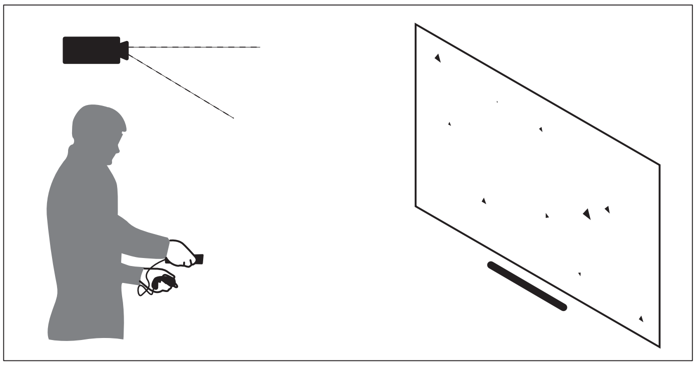

Figure 1: Standard set up of Sputnik.

5.2. The Virtual Scene

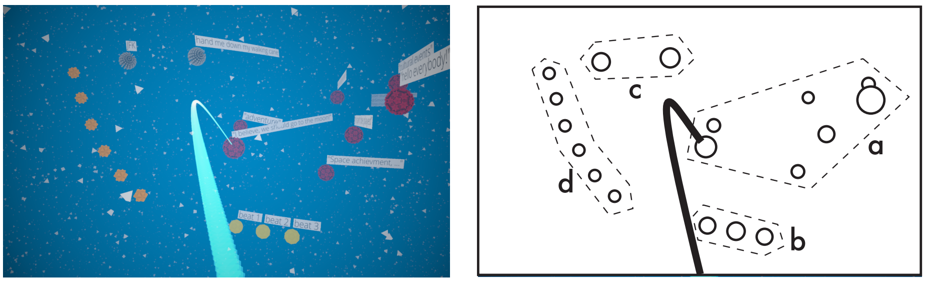

Figure 2: The interactive objects in the scene: (a) samplers, (b) players, (c) tape machines, (d) harmonic harp

Figure 2 shows an overview shot of Sputnik. The scene in Sputnik is filled with a randomly generated star field, that serves not only an aesthetic purpose but also provides important reference points for the user's orientation. Inside the star field exist 4 classes of objects, the user can interact with: sampler (red), player (yellow), tape machine (grey) and harmonic harp (orange). All but the last also sport textual labels, containing a short text or name describing the interactive object as it can be seen in figure 3. To provide depth cues to the users, the scene is additionally filled with a homogeneous, coloured fog that causes objects to fade in the distance.

The purpose of the visual representation is twofold. On the one hand it is the sole graphical interface for the user and on the other hand it should be visually pleasing

Figure 3: A group of samplers and the close up of a single sampler with their attached labels.

for the audience and allow them to understand the process of music creation in a live setting. Common music software usually focuses on the performer, leaving the visual performance in most cases to a dedicated and specialised VJ. Sputnik tries to bridge this gap by using the visual representation both as an interface to the performer and the audience.

The textual labels in Sputnik (figure 3) are aimed at both the performer and the audience. The text on the labels is fixed and does not change during the performance. It can be used to convey additional information to the performer about a certain object, but it can also allow the audience to gain more information about the piece. The function of the objects can become clear even if the object is not currently in use. This can also introduce an element of anticipation when an object with a certain label is visible but the performer does not yet interact with it.

5.3. Navigation and Camera Controls

Sputnik is controlled from a first person perspective. The user can navigate the scene by pushing the Nunchuck's analogue stick. Pushing the stick in the desired direction moves the camera accordingly.

Tilting and panning is controlled by pointing the Wiimote to the top/bottom/left/right of the screen. The farther it is pointed away from the neutral center position the faster the camera movement is. Figure 4 illustrates the navigation.

Sputnik's camera uses a fixed *up direction*. The fixed up direction is commonly seen in cinema and video games, even though it is not intrinsic in the space inspired setting. It should furthermore limit the camera's degree of freedom and make it more accessible.

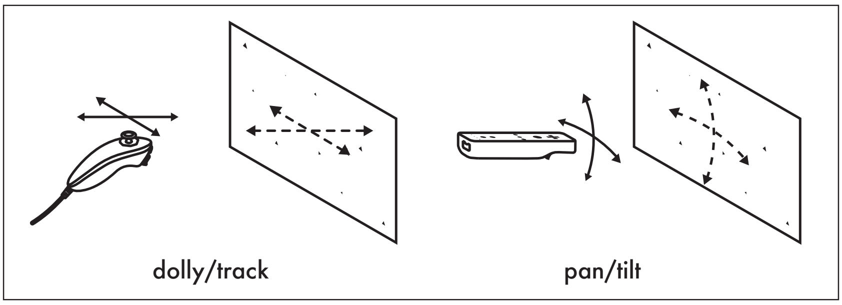

Figure 4: The Nunchuk's analogue stick controls the camera's dolly and track movements, the Wiimote's IR pointer controls the camera's tilt and pan movements.

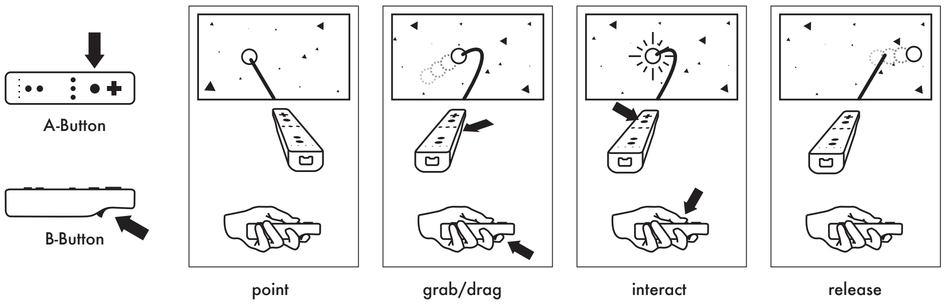

Figure 5: The basic interaction vocabulary of Sputnik

5.4. Interaction

Figure 5 illustrates the basic interaction vocabulary of Sputnik. The user can interact with the 3D scene via an elastic *arc of light* metaphor that is intended to act as a bodily extension of the user into the virtual space. Through this arc, the user can *point* at objects, *grab* them and also *drag* them around the scene.

Interactive objects behave in a simplified, physically plausible way, each featuring distinct weight and friction. Dragging objects causes the arc to bend like a fishing rod, reflecting the physical properties of the object.

Each interactive object in the scene can react individually to user interaction. The following classes of objects currently exist in Sputnik (see figure 2 for an overview):

Sampler (red) The sampler object reacts to the press of the A-button while it is grabbed. While the button is held a preloaded sample is played in a loop and stops immediately when the button or the object is released. Playback always starts at the beginning of the sample.

Player (yellow) The player object reacts to the press of the A-button while it is grabbed. Pressing the A-button starts and stops the playback of a preloaded sample. It is not automatically stopped when it is released. Playback is looped and always starts at the beginning.

Tape Machine (grey) Modelled after an old tape machine and inspired by musique concrète, the tape machine object controls the play back speed of a preloaded sample via the object's movement speed in the virtual space. The faster it moves the faster the playback. Playback is looped.

Harmonic Harp (orange) The orange spheres together form a kind of harmonic harp. Each sphere controls the volume of a single sine oscillator. Each sphere also has its own origin towards which it is dragged all the time, causing the spheres to swing in pendulum like motions if moved. The distance from the sphere's center to this origin determines its volume and the underlying oscillators are tuned to the natural harmonic series, which makes them easy to play with and avoids dissonances.

Each interactive object is preloaded with a different sample or sound in an external application. By interacting with the different objects and arranging them in the virtual scene, users can create musical performances that are both musically and visually expressive.

5.5. Simplified Physics

Sputnik uses simplified physics simulations to give the interactive objects their distinct physical properties. An object's properties are described by *mass*, a *movement vector*, as well as a movement *damping constant*.

The dragging force of the arc on an object is linear in the distance from the object to the intersection of the unbent arc with the plane that goes through the object, with a normal pointing at the camera. This intersection point also defines the direction of the force vector.

In each frame, the speed of an object is then computed as

$$v_i = v_{i-1} + \frac{F * \Delta t}{m} \tag{1}$$

where v_i is the object's movement vector in frame i, F is the current force, m is the mass and Δt is the frame time. Force is only applied at the center of the objects, and thus never causes a rotation of the object.

Damping of the object's movement is implemented by a simple exponential decay, given as

$$v_i = v_{i-1} * e^{-\lambda * \Delta t} \tag{2}$$

where λ is the damping constant.

6. Implementation

Sputnik is implemented in C++ with Mac OS X as its development platform. The project is split into two sub projects: The special purpose Sputnik and the more general purpose *kocmoc-core*. Some parts of kocmoc-core existed prior to this project (about 50 working hours worth of code), but most of it was created in the course of the project.

The source code of both Sputnik and kocmoc-core is licensed under the permissive MIT open source license and is publicly available on github[5] [6].

Sputnik uses OpenGL 2.1 with a few extensions to display the virtual scene. The renderer is fairly simple and displays unlit, textured objects with baked ambient occlusion maps. The computation of the homogeneous fog is realised in the vertex and fragment shader and a post processing effect is applied. This effect adds a barrel distortion to compensate the wide field of view and vignetting. A simple form of full screen anti aliasing is also added in this stage.

The *Assimp*[7] and *devIL*[8] libraries are used to load assets. Font rendering is implemented using the *freetype*[9] library. *RtMidi*[10] is used to send MIDI messages to external applications and *WiiC*[11] was used to interface with the Wiimote and Nunchuck controller.

[5] `https://github.com/SimonWallner/sputnik`
[6] `https://github.com/SimonWallner/kocmoc-core`
[7] `http://assimp.sourceforge.net/`
[8] `http://openil.sourceforge.net/`
[9] `http://www.freetype.org/`
[10] `http://www.music.mcgill.ca/ gary/rtmidi/`
[11] `http://wiic.sourceforge.net/`

Figure 6 illustrates the data flow in the main run-loop. First, the devices are polled, which causes input callbacks to be fired. The components are subsequently updated and finally the scene is sent to the renderer.

6.1. Creating sound

Sputnik's interactive objects individually react to the users input and use the MIDI protocol to communicate with an external application. Sputnik itself does not create any sounds. Sound creation is handled entirely by an external application.

Thanks to the simplicity and the pervasiveness of this protocol, virtually every music software can be used together with Sputnik. In the current set up *pure data (pd)*[12] is used for sound creation. A simple patch is used that builds on *boctok-1*, a small collection of pd patches written by the author prior to this project.

Communication is only one way: From Sputnik to the external application. The MIDI channel and controller number every object uses can currently only be changed in the source code.

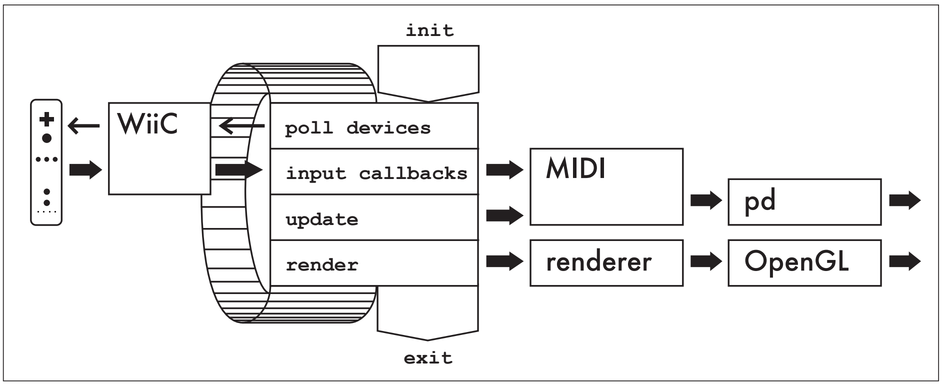

Figure 6: The data flow in the run-loop.

7. Wiimote Input

The Wiimote controller was chosen for its pointing functionality as well as the motion sensing capabilities, even though they haven't been used in the final project. The accompanying Nunchuck controller was also used to allow the standard, analogue stick driven navigation.

[12] http://puredata.info/

The Wiimote has a built in low resolution IR camera and its feed is directly processed on chip. The feed itself is not accessible, but data of up to 4 tracked IR points can be read. the Wiimote's update rate is reported to be $100Hz$ which gives a lower bound for the worst case lag of $10ms$. The used Wiimote library takes care of most of the processing and conveniently returns the computed pointer location in relative screen coordinates.

A drastic constraint of the Wiimote's IR pointer however is the relatively narrow field of view. It is very easy to leave the small sensing area. During the development it turned out that using the Wiimote for the tilt/pan movements of the camera can alleviate this problem a little. Users seem to intuitively try to compensate the camera's rotation, thus maintaining a focus on the neutral area in the center of the screen. Without it users easily lost the focus an had troubles finding *back onto the screen*.

The used Wiimote library supports more than one Wiimote and the input system in Sputnik would also support multiple input devices at the same time. The option to dual wield two Wiimotes and to interact with two arcs of light simultaneously was given up in favour of the more standard analogue stick track/dolly controls, that allow the users to easily navigate the scene.

8. The Arc Of Light

The arc of light forms the foundation of Sputnik. It acts as a bodily extension of the user into the virtual scene. Through the arc, the user can use the interaction vocabulary described above. It has the following behaviour.

If no object is grabbed, the arc follows the user's input directly. This input is not filtered and is used directly *as is*. This introduces a slight jitter but on the other hand does not take away from the system's responsiveness.

Low pass filtering however is desired in most scenarios and is indirectly implemented via the physical properties of the interactive objects. The higher the weight of the object for instance, the stronger the filtering is. While grabbing an object the arc bends accordingly to the physical properties of the object and the dragging force of the user. It bends like a fishing rod, a natural metaphor that seems to be readily understandable to the users.

One of the core advantages of the arc is that even very strong filtering can be achieved without making the system feel laggy and slow. It happens in a way that is transparent and understandable for the user, and in the user study the participants reported that the system is very responsive.

9. Evaluation

9.1. Method

Sputnik was evaluated in a user study where participants were asked to perform certain tasks, followed by a questionnaire and an interview. Sputnik was tested with only one person at a time.

A relatively wide range of questions and topics was used in the user study to gain a broad overview of how Sputnik is used, what general problems arise, and how suitable it is as an interface for musical expression. The user study was recorded on video and the interview was audio recorded for later analysis.

9.2. Participants

Musical experience was a criterion in finding participants and from the initial 6 participants, three unfortunately cancelled the test on short notice. The three remaining participants (all male) had an average age of 26 (SD = 5.2) and all had a musical background having played classical instruments but not on a professional level. Two had additional experience with computer and sample based music. The participants were not compensated for taking part in the user study.

None of the participants had used a Wiimote prior to the user study and all were right handed and colour normal. Two participants have seen short explanatory demo videos of Sputnik that were used in order to find participants.

9.3. Setup

The user study was conducted in a darkened room with the participant standing about 4 meters away from the overhead projected screen with a diagonal of about 2 meters. The screen had an aspect ratio of about 16:9 and the screen's center was about 1.5m above ground. The Wiimote sensor bar was positioned a few centimetres below the lower edge of the picture. A small stereo speaker set was used that was located on the floor right behind the participant.

9.4. Procedure and Design

At the beginning of the user study, participants were asked for basic personal data. The participants were informed about the user study and the controls were briefly explained.

The first task was to try out the navigation and familiarise with the controls. This included basic navigation and orientation as well as grabbing and interacting with objects.

For the second task, participants were asked to describe for every class of interactive objects, what they do and how they react to user interaction. This should make sure that the participants fully explore the system before continuing, and also tested if the behaviour of the objects is understandable.

The third task consisted of a brief improvisational musical performance. Time was given to plan and structure it in advance and participants were asked to start when they felt ready. The length of the performance was not specified, only a vague rule of *not too short, not too long* was given. This test should evaluate Sputnik's potential for musical expression, and also force a more goal oriented use of Sputnik.

The user study was concluded with a brief ordering task. The participants were presented with four objects and were asked to bring them in order according to their weight. Those objects did not create sounds. With this test the perception of the weight of interactive objects was tested.

After the practical part, a one-page questionnaire was given to the participants. It consisted of 4 Likert-items and 7 QUIS[Chin et al., 1988] questions, 4 of them taken from the original paper and 3 additional questions to capture aspects not found in QUIS. A short interview, guided by a set of pre-written questions was conducted afterwards. The questionnaire and interview form can be found in appendix A and B.

Due to the small sample size, no quantitative analysis was performed on the Likert and QUIS items. Notes were taken during the user study and interview and all video and audio material was reviewed to get a general overview. As the analysis was narrowed down, this material was consulted repeatedly to confirm and underpin the findings.

9.5. Results

Overall, reactions to Sputnik were rather positive, which is hinted in the results of the QUIS inspired questionnaire seen in figure 7. Due to the small number of participants, no direct and strong conclusions can be drawn from this evaluation. It shall rather be seen as a precursor to a fully fledged user study and as a hint as to what Sputnik does good, and where there is room for improvement.

Unfortunately, the results of the Likert items were inconclusive: The first question, "Sputnik is easy to use" resulted in the answer triple (strongly disagree, neutral, agree) and all other items resulted in (disagree, neutral, agree). The results to the QUIS questions in figure 7 lack significance, but show a mildly positive trend. Only the question about responsiveness has unanimously received high values, marking it as a strength of Sputnik.

The following sections break down the findings into *Controls and Navigation*, *Visual*, *Interation*, *Improvisational Performance* and finally the object sorting task is reviewed. The break down into this categories is not definitive and the overlapping borders are rather fuzzy than strict.

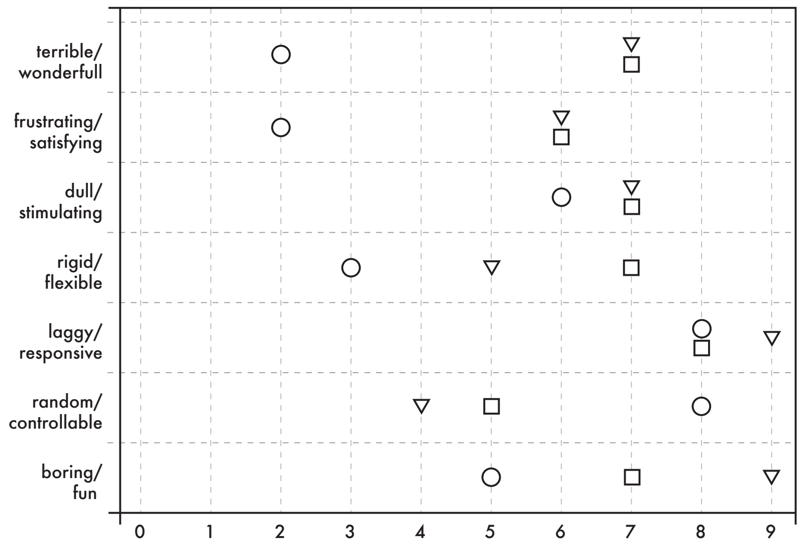

Figure 7: QUIS results...

9.5.1. Controls and Navigation

Even though none of the participants had prior experience with the Wiimote, all managed to effectively use Sputnik during the user study. After only a few moments the participants were able to navigate the scene and interact with the objects.

A problem, however, was the narrow field of view (FOV) of the Wiimote, resulting in uncontrollable camera movement if the sensor bar is not in the Wiimote's FOV. During the test this happened about 3-5 times per participant.

Another problem was insufficient pointer precision. All participants struggled at least once in trying to grab small or faraway objects. This might be influenced by the slight jitter of the Wiimote's pointer and its narrow FOV, requiring rather subtle movements.

Differences in movement patterns could also be observed. Two of the three participants did not move the camera very often. Phases of interaction (rotate, grab, interact) and camera movement were clearly separated and camera movement was rather erratic. The third participant also begun in a similar fashion, but then gradually began to combine movement and interaction into more sophisticated movements like *circle strafing* (moving sideways while turning) and moving while dragging an object.

All participants tried to pull objects by grabbing them and moving backwards at the same time. This is not implemented in Sputnik and this was noticed and explicitly mentioned by all participants during the test.

9.5.2. Visual

All participants were missing feedback of the interactive objects regarding their musical state, e.g., a way to see if a player object was playing or not. One participant mentioned explicitly that the visual representation has no meaning, stating that: "[...] objects and their forms, and so on..., didn't make any sense". Answering another question he also mentioned that it would be more interesting if also the star field was interactive and reacted to the music performed.

The participants seemed to not pay much attention to the textual labels of the objects. The labels have a fixed orientation and the camera was often in a position from which they were not readable. This can be related to the rather simple navigation patterns mentioned above and that the effort required to move the camera in a position from which the labels were readable was to high.

9.5.3. Interaction

All participants could deduce the functions of the sampler and player objects. Descriptions of the tape machine and harmonic harp were not as precise, but captured their general mode of operation.

Participants seemed to have no problems using the arc and all participants responded positively when being asked about immersion into the scene. One participant even said at the end that: "I liked the link from me to the objects, when you hold the objects. It made sense". A point noticed and mentioned by all participants was the arbitrary distance limit of the arc. Objects could only be grabbed if they were inside a certain range.

Two of the 3 participants also tried to waggle the Wiimote, a gesture which is not directly used in Sputnik.

In regards to the samplers, all three participants indicated the need for a way to quickly and directly access some of the objects, to either have certain sounds available all the time, or to quickly change between sounds.

Even though Sputnik was not running at a very high frame rate during the test ($\sim 30 - 40$fps) participants indicated in the QUIS that it was *very responsive* and no comments about lag were made in the interview.

9.5.4. Improvisational Performance

All participants used a similar pattern for their improvisation, alternating between the different groups of objects. The participant without prior experience with computer and sample based music had trouble putting together a coherent performance and seemingly disliked the task.

The two users with experience in computer and sampler base music, arranged the objects before the performance and seemed to be *planing ahead*. A procedure that is probably similar to how they would approach such a task with their usual setup and tools.

Likeability of the interactive objects seems to correlate with the amount of interactivity they provide and how it is used in the system. The tape machine and harmonic harp generally received better comments than the sampler and player objects. Two participants said that they disliked the player object for being uninteresting and lacking interaction.

All participants expressed the need to exchange the sounds and fit Sputnik to their needs in order to be useful in a live setting.

One participant reported fatigue in his shoulders, an aspect of Sputnik that was not accounted for in the user study, but is an important factor in real world usage scenarios.

9.5.5. Object Sorting

Two participants sorted the objects correctly, one brought the lightest two in the wrong order. All participants used a continuous grab to assess the weight of the objects. During this task, all participants made comments about the physicality of the objects like: "This is harder to drag. This is less hard. This is quite easy", "B is lighter than C, it can be moved more easily."[13], "C is arguably the heaviest."[14].

During the development, another user tried to throw the objects to observe their weight, but this behaviour was not observed in the user study.

10. Discussion and Conclusion

The user study indicates that Sputnik is relatively easy to pick up and use, even for users with no prior experience with the Wiimote controller. More sophisticated and advanced movement patterns seem to emerge as the participants grow accustomed to the system.

What the user study also shows is that the Wiimote is not very suitable as a controller for Sputnik, due to its narrow field of view. It is too easy to move out of the FOV resulting in a loss of control over the camera. Users could usually recover within a few seconds, but still this would be a show stopper if used in an actual live

[13]translated from German: "B ist leichter als C, das lässt sich leichter bewegen."
[14]translated from German: "...ist C wohl am schwersten."

performance. Other input devices like the PlayStation Move[15] controller, Microsoft's Kinect[16], or handheld devices equipped with motion sensors like modern smartphones or gyroscopic mice should be evaluated as alternatives.

The arc of light worked well in providing a bodily extension of the user into the virtual scene. It can be effectively used in combination with physical objects to low pass filter user input without loosing the perceived responsiveness of the system.

The question of whether musical and visual expressiveness can be achieved through Sputnik can not be answered in this evaluation. More time needs to be given to the participants and more possibilities to customise Sputnik to their needs and musical and visual style. An approach similar to the one used in [Gurevich et al., 2010] could be a good starting point for further evaluations. In this work, users were given a highly constrained one button instrument that could only create a single tone. The participants had a week to practice and prepare a performance, resulting in diverse play styles. Even though Sputnik is far less constrained, approaching it from this angle could yield interesting insights into its expressive potential. Sputnik's constraints could be a source of creativity rather than a hindrance, as it is mentioned in [Magnusson, 2010].

Even though Sputnik is not a TUI, a strong relation to this field exists. Sharlin et al. [2004] introduced the concept of *spatial TUIs*, TUIs that feature a strong spatial component. Sharlin et al. elaborate on *I/O unification*, the tight coupling and unification of the action and perception space, an aspect that can clearly be seen in Sputnik's interactive objects.

The tape machine and harmonic harp in Sputnik employ strong I/O unification. Input and output are tightly coupled through their virtual representation and cannot diverge. Furthermore, their spatial configuration is directly mapped to the sound parameters.

By seeing Sputnik as a *virtual TUI*, with the arc of light as a bodily extension of the user into the virtual space, existing TUI literature can be used to study and evaluate Sputnik. Conversely, translating actual TUIs into a virtual environment can open up new viewing angles on them.

11. Future Work

This project raised a lot of questions and opened up many directions for future work:

Sputnik's impact on a live audience and how it can be used for visual expression should be explored. This is an important aspect of Sputnik and a larger scale study is needed.

To be useful for artists and users with less of a technical background, editing tools

[15] `http://en.wikipedia.org/wiki/PlayStation_Move`
[16] `http://en.wikipedia.org/wiki/Kinect`

should be provided to allow the artist to change the scene, create new objects and models and change the input/output mapping. Additionally, the performance should be increased to run at > 60fps for a smoother experience.

The Wiimote's motion sensing capabilities were left untapped in Sputnik. Some participants in the user study tried to waggle the Wiimote, a strong indication that more advanced motions and gestures should be explored. Two handed interaction with a second Wiimote instead of the Nunchuck controller could also be interesting to evaluate. It would allow two handed interaction and more advanced gestures known from touch devices like pinching and rotating.

Direct visual feedback of the interactive objects to the generated sound should be implemented. The lack of it was mentioned by all participants and it might also be crucial for the effect on a live audience. Additionally, the musical possibilities should be explored by devoting the time and creativity to create genuine pieces of music for Sputnik and to perform them live.

The arc of light metaphor should be further explored in other fields outside the domain of musical interfaces. It could be used to low pass filter user input while still maintaining a high perceived responsiveness of the system, or to add a sense of depth to an otherwise two dimensional input. Using the arc as a bodily extension can benefit VR applications and provide a good link between the user and the virtual environment.

12. Acknowledgements

I would like to express my greatest thanks to my supervisor Esben Warming Pedersen, who always turned me in the right direction and helped me in finding and refinding the project's focus. Great thanks also to the testers of Sputnik who provided me with many important clues and insights.

References

Chin, J. P., Diehl, V. A., and Norman, L. K. (1988). Development of an instrument measuring user satisfaction of the human-computer interface. In *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '88*, pages 213–218, New York, New York, USA. ACM Press.

Cook, P. (2001). Principles for designing computer music controllers. In *Proceedings of the 2001 conference on New interfaces for musical expression*, NIME '01, pages 1–4, Singapore, Singapore. National University of Singapore.

Dobrian, C. and Koppelman, D. (2006). The E in NIME: musical expression with new computer interfaces. In *Proceedings of the 2006 conference on New interfaces for musical expression*, pages 277–282.

Fels, S. and Lyons, M. (2011). Advances in new interfaces for musical expression. In *ACM SIGGRAPH 2011 Courses*, pages 1–169, Vancouver, British Columbia, Canada. ACM.

Fitzmaurice, G. W., Ishii, H., and Buxton, W. A. S. (1995). Bricks. In *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '95*, pages 442–449, New York, New York, USA. ACM Press.

Gurevich, M., Stapleton, P., and Marquez-Borbon, A. (2010). Style and Constraint in Electronic Musical Instruments. In *Proceedings of the 2010 conference on New interfaces for musical expression*, number Nime, pages 106–111, Sidney, Australia.

Gurevich, M. and Treviño, J. (2007). Expression and its discontents. In *Proceedings of the 7th international conference on New interfaces for musical expression - NIME '07*, page 106, New York, New York, USA. ACM Press.

Ishii, H. and Ullmer, B. (1997). Tangible bits. In *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '97*, pages 234–241, New York, New York, USA. ACM Press.

Jordà, S., Geiger, G., Alonso, M., and Kaltenbrunner, M. (2007). The reacTable: Exploring the Synergy between Live Music Performance and Tabletop Tangible Interfaces. In *Proceedings of the 1st international conference on Tangible and embedded interaction - TEI '07*, page 139, New York, New York, USA. ACM Press.

Kiefer, C., Collins, N., and Fitzpatrick, G. (2008). Evaluating the wiimote as a musical controller. *Proceedings of the International Computer Music Conference*, pages 17–17.

Magnusson, T. (2010). Designing Constraints: Composing and Performing with Digital Musical Systems. *Computer Music Journal*, 34(4):62–73.

Mäki-Patola, T., Laitinen, J., Kanerva, A., and Takala, T. (2005). Experiments with virtual reality instruments. In *Proceedings of the 2005 conference on New interfaces for musical expression*, pages 11–16. National University of Singapore.

Miller, J. (2010). Wiiolin: a virtual instrument using the Wii remote. In *Proceedings of the 2010 conference on New interfaces for musical expression*, number June, page 497ff, Sidney, Australia.

Miyama, C. (2010). Peacock : A Non-haptic 3D Performance Interface. In *Proceedings of the 2010 conference on New interfaces for musical expression*, number Nime, pages 380–382, Sidney, Australia.

Patten, J., Recht, B., and Ishii, H. (2002). Audiopad: a tag-based interface for musical performance. In *Proceedings of the 2002 conference on New interfaces for musical expression*, pages 1–6, Dublin, Ireland.

Pedersen, E. W. and Hornbæ k, K. (2009). mixiTUI. In *Proceedings of the 3rd International Conference on Tangible and Embedded Interaction - TEI '09*, page 223, New York, New York, USA. ACM Press.

Rodet, X., Lambert, J.-P., Cahen, R., Gaudy, T., Guedy, F., Gosselin, F., and Mobuchon, P. (2005). Study of haptic and visual interaction for sound and music control in the phase project. *Proceedings of the 2005 International Conference on New Interfaces for Musical Expression (NIME05)*, pages 109–114.

Shaer, O. (2009). Tangible User Interfaces: Past, Present, and Future Directions. *Foundations and Trends® in Human–Computer Interaction*, 3(1-2):1–137.

Sharlin, E., Watson, B., Kitamura, Y., Kishino, F., and Itoh, Y. (2004). On tangible user interfaces, humans and spatiality. *Personal and Ubiquitous Computing*, 8(5):338–346.

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

Wanderley, M. M. and Orio, N. (2002). Evaluation of Input Devices for Musical Expression: Borrowing Tools from HCI. *Computer Music Journal*, 26(3):62–76.

Zappi, V., Italiano, I., Brogni, A., and Caldwell, D. (2010). OSC Virtual Controller. In *Proceedings of the 2010 conference on New interfaces for musical expression*, number Nime, pages 297–302, Sidney, Australia.

Appendices

A. User Study Questionnaire

Sputnik User Study Evaluation Form

Simon Wallner

December 12, 2011

1 Statements

For the following statements, please indicate your level of agreement.

Sputnik is easy to use.

strongly disagree	disagree	neutral	agree	strongly agree
○	○	○	○	○

The interface felt natural.

strongly disagree	disagree	neutral	agree	strongly agree
○	○	○	○	○

I could feel the weight and physical properties of the objects.

strongly disagree	disagree	neutral	agree	strongly agree
○	○	○	○	○

Using Sputnik could improve live performances.

strongly disagree	disagree	neutral	agree	strongly agree
○	○	○	○	○

2 Adjectives

For the following adjectives, please specify how they apply to Sputnik.

Overall, Sputnik is...

terrible wonderfull
0 1 2 3 4 5 6 7 8 9

frustrating satisfying
0 1 2 3 4 5 6 7 8 9

dull stimulating
0 1 2 3 4 5 6 7 8 9

rigid flexible
0 1 2 3 4 5 6 7 8 9

laggy responsive
0 1 2 3 4 5 6 7 8 9

random controllable
0 1 2 3 4 5 6 7 8 9

boring fun
0 1 2 3 4 5 6 7 8 9

B. User Study Evaluation Form

Sputnik user study – evaluation form

1 Data

Name ______________________________

Participant code/time ______________________________

Age ______________________________

Gender ______________________________

Educational background ______________________________

Musical background ______________________________

Exp. w/ virtual env. ______________________________

Wiimote exp. ______________________________

left/right handed ______________________________

colour normal ______________________________

2 Tasks

1. Take the controller and familiarise yourself with the controls. Try to navigate the scene. Use the *Wiimote pointer* to look around, the *analogue stick* to move, the *B button* to grab objects and the *A button* to interact with certain objects. We will continue, when you feel ready. Press the *home* button to unlock.
2. Explore the objects in the scene. For each class of objects, please tell me what these objects do, to what interactions they react, and how they influence the sound.
3. Now I would ask you, to give a short impromptu performance. You can choose the length of the performance but it should neither be too long nor to short.
4. Please sort these 4 objects according to their weight, from the lightest to the heaviest.

3 Notes

3.1 Basic Navigation

3.2 Models and Metaphors

3.2.1 Sampler (red)

3.2.2 Player (yellow)

3.2.3 Tape Machine (grey)

3.2.4 Harmonic Harp (orange)

3.3 Performance

3.4 Sorting Task

4 Interview

4.1 Performance

1. How was the performance? Are you content with it? Did you enjoy it?

2. What was the hardest thing to achieve in your performance?

3. What would you want to improve?

4. What limited/constrained you the most?

4.2 Navigation and Control

1. What was the hardest thing about the navigation?

2. How could the controls be improved?

4.3 Visual Representation & Immersion

1. How would using Sputnik impact a live performance?

2. Did you feel immersed into the scene?

4.4 Music & Sound

1. Which object did you particularly enjoy?

2. Which object did you least enjoy?

3. What would be the *natural* application of Sputnik (exploration/composition/live/exhibition)

4. Where would you put Sputnik in your creative process?

5. How would you compare it to other systems/tools/input devices?

6. What possible further musical uses could you envision?

4.5 Misc

1. What other uses could you envision?

2. Is there anything else you would like to add?