YouHero - Making an Expressive Concert Instrument from the GuitarHero Controller

Palle Dahlstedt
Dept. of Applied IT
Univ. of Gothenburg, Sweden
palle.dahlstedt@gu.se

Patrik Karlsson &
Katarina Widell
Duo EN.D.E
info@patrikkarlsson.se

Tony Blomdahl
Geiger Musik
tony.blomdahl@gmail.com

ABSTRACT

The idea behind the YouHero was two-fold: First, to make an expressive instrument out of the computer game toy guitar controller from the famous game GuitarHero. With its limited amount of control parameters, this was a challenge. Second, we wanted to provide an alternative to the view that you become a hero by perfect imitation of your idols. Instead, play yourself. You are the hero. In this paper, we describe the design of the instrument, including its novel mapping approach based on switched timbre vectors scaled by accelerometer data, unconventional sound engines and the sound and mapping editing features, including manual editing of individual vectors. The instrument is evaluated through its practical applications during the whole project, with workshops with teenagers, a set of state-funded commissions from professional composers, and the development of considerable skill by the key performers.

Keywords

gestural interface, guitar hero, mapping, synthesis, music pedagogy, performance, improvisation

1. INTRODUCTION

The YouHero project was initiated in 2009 by Patrik Karlsson and Martin Q Larsson, inspired by the electronic instruments developed by Palle Dahlstedt, and by his ideas about electronic improvisation. The goal was to make a concert instrument based on the Guitar Hero controller, expressive enough to be used as a concert instrument, but also easy enough to learn, so that it can be played and written for by school children in workshops. In the Guitar Hero computer game, you are trained in imitating your idols. Instead, we wanted to inspire users to really play music. Hence the name of the project – you are the hero. The project has included a series of workshops with school kids, but also several concerts at contemporary music festivals.

1.1 System overview

The YouHero instrument consists of four main parts: the physical controller, a mapping engine, a pitch engine, and a couple of sound engines (see Fig. 1). It uses an Xbox 360 GuitarHero controller, providing a set of performance controls (neck switches, plectrum switch, whammy bar, accelerometers) and meta controls (octave select, timbre push and pop, preset selection). Through USB, it communicates with a laptop running custom software, implemented in Max, Java and Synth-Maker (now FlowStone), all contained in a Max wrapper handling control data parsing, data flow, user interaction (dialog boxes) and sound streams. The mapping engine is implemented in a Java object inside Max, and the sound engines are implemented in SynthMaker, compiled into VSTi plugins, hosted by Max. The whole package was compiled into a runtime application.

In this paper we concentrate on the technical and design aspects and implementations, and some preliminary discussion about the pedagogical ideas behind the instrument and the musical results.

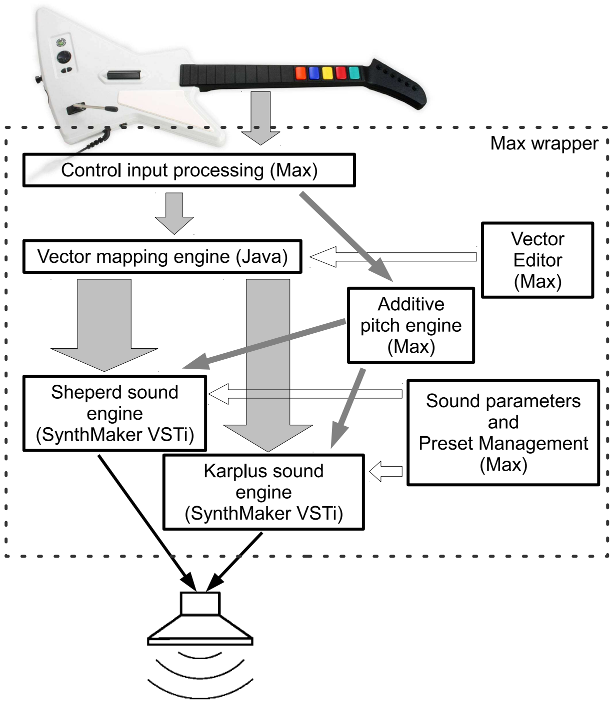

Figure 1: An overview of the YouHero instrument, consisting of a standard GuitarHero controller and custom software.

1.2 Previous work

There are some previous attempts at using the Guitar Hero controller as an instrument for conventional rock, e.g., the Guitar Zeros band [1], using an approach quite similar to ours, but with less focus on expressivity and novelty. It has also been used as controller in a system for music therapy, using custom sound and playback engines [5], providing simplified interaction for disabled players. A related project uses the internals of the Guitar Hero as a platform for a game controller based on muscular signals, for amputees[7].

In contrast to these projects, our goal was to make an expressive, experimental concert instrument to be played by both kids and professional musicians. By expressive, we mean nuanced control of dynamics and timbre, and access to a wide variety of timbres.

2. IMPLEMENTATION

The software is developed in Max (general framework, in/out handling), SynthMaker (sound engines) and Java (mapping computation).

2.1 Interface and mapping

The game guitar controller, connected through USB to a Windows Laptop, has five switches on the guitar neck, and a plectrum switch in the middle, a two-axis accelerometer and a whammy bar (bend arm). Control data is accessed through the Max *hi* object. The five colored neck switches are used for pitch control, while the plectrum switch triggers notes. The three continuous sensors are essential for expressive playing. The whammy bar is used for pitch bend, while one accelerometer dimension is used to detect tilt, i.e., how much the neck is raised. This feature is used as a primary control parameter, scaling dynamics and timbre vectors, which fits the look and feel of the instruments – this is indeed an archetypical rock gesture. The second accelerometer dimension indicates tilt along the central axis of the guitar. This was used for minute modulation of synthesis parameters, giving life to the sound. For example, it can be used to produce a vibrato or tremolo by shaking the guitar up and down. The accelerometer mappings encourage interesting theatrical performance qualities.

Each neck switch activates a vector in synthesis parameter space, and several switches can be pressed at once. Active vectors are scaled by the tilt angle of the guitar and summed, to calculate the current synthesis parameter set. If an interesting sound is found, the whole vector mechanism can be transposed to the current position, altering the characteristics of the instrument. These transpositions are pushed to a stack and can be popped or reset to the original origin at any time, using buttons on the guitar controller.

This vector mapping is a novel and somewhat different implementation of a previously developed algorithm [3]. It was a challenge to implement it for a more limited control interface, where most of the playing is done with a small set of switches. The tilt mapping to scale amplitude and vector length simultaneously turned out to be a good solution, providing enough control and expressivity for normal playing. In the other instruments based on this vector mapping approach, the vectors have been randomized at design time. However, in this instrument they can be either randomized or hand-tuned by ear. This is possible since there are not so many degrees of freedom, and the number of vector components (8) is manageable. The software provides an interactive vector editor where random vectors can be tested, or specific values for the vector components can be entered with the mouse (Fig. 2).

2.2 Sound Engines

There are two different sound engines, called Sheperd and Karplus, each with a set of synthesis parameters, modulated by the output of the vector mapping engine. The Sheperd sound engine is based on intermodulating FM synthesis, with some unusual features. Since harmonic FM synthesis is dependent on integer frequency ratios between oscillators, a particular system was used to ensure smooth transitions between different integer ratios. It has two oscillators, each modulating the other, providing potential for chaotic results, depending on modulation strengths [2]. Each oscillator consists of a bank of sub-oscillators tuned in octaves ($2^n f, n = 1, 2...$) in one case, the harmonic series (nf) in the other. As the pitch is modulated by values from the mapping engine, the sub-oscillator bank is scanned (with cross-fade). The fundamental pitch does not rise, but instead the oscillator smoothly enters a higher register. In this way, there is always an integer ratio between the actual output of the two oscillators. There is still potential for noise and complex sounds, depending on the FM strengths. The range of those can be set globally, and stored in a sound preset, allowing for some control of the complexity or harshness of the sound.

The Karplus sound engine is based on Karplus-Strong string models and emulated room feedback, essentially emulating a guitar, the amp and the room around it at the same time. Because of the nonlinear feedback paths, including a reverb model to emulate the room, pitch is sometimes difficult to control, just like a real guitar played with speaker cabinet feedback. Essentially, it eliminates the loud process of feedback from guitar amplifier through the guitar microphones while keeping the sonic results and the nonlinear pitches that result from it.

2.3 Pitch engine

Pitch is controlled through an additive system, where each neck switch corresponds to an interval of 1, 2, 4, 7 and 12 semitones. Simultaneously pressed switches add up, and the component intervals were chosen based on a decomposition of common diatonic scales and harmony (minor second, major second, major third, fifth and the octave). An interval set based on 2^n semitones would allow for more variety, but would make less sense musically, and common intervals would be difficult to play. Similar pitch mappings have been used in other electronic instruments [3].

The sum interval is added to a base note, which can be set differently for each preset, and the range controlled with octave selectors on the controller.

Playing the neck switches affects both pitch and timbre, which might be perceived as a limitation. However, this is how acoustic instruments work [4], where different parameters are coupled to each other. E.g., each note of a clarinet has a different timbre, because of changing ratios between the pitch and the fixed shape of the instruments causing formant-like filtering, etc.. For most notes, there are several neck switch combinations to produce them, so it is in theory possible to even make a timbre trill.

With the choice of sound engine, a number of sound parameters available for tweaking, and the ability to hand-tune the vectors, a wide range of sounds and sonic behaviors can be produced. To take advantage of this, the user can store ten presets, to be recalled during a performance. In the preset, all parameters visible in the main dialog and in the vector editing dialog are included.

3. PERFORMANCES AND WORKSHOPS

Four works were commissioned for the YouHero instrument, funded by the Swedish Arts Council through a series of commission grants. These compositions ranged from game pieces (see Fig. 3 for a score fragment) to noise quartets[1], where loudness and stage appearance was in focus, rather than the details of the musical structure. One composer (Lisa Ullén) made very intimate duets for recorder and

[1] http://www.youtube.com/watch?v=_gydrs6hs9o

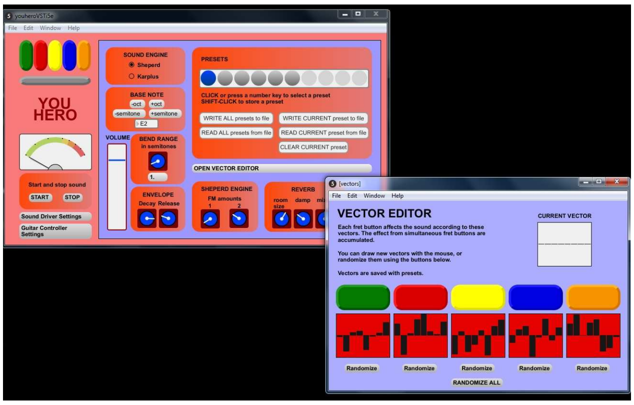

Figure 2: A screenshot of the YouHero software user interface. To the left is the main dialog box, with VU meter, general sound controls, and access to various settings. It also contains some sound parameters, as well as sound preset management. To the right is the vector editor dialog box, where the user can hand-tune the underlying vectors of the mapping engine.

YouHero guitar. Finally, one composer (Lars Bröndum) composed really detailed written scores and provided custom presets. He also made a series of requests to the software designer (PD), to make his ideas possible to realize. This relates to the conclusions by McPherson and Kim [6] regarding the issue of creating a community around a new instrument. In their case study, there were several requests for a relaxation of constraints, to accommodate the wishes of the composers. This is probably a natural part of the evolution of musical instruments, and the interplay between musician and instrument maker.

The commissioned pieces were performed in a series of concerts around Sweden, together with group improvisations on the instruments. One other contemporary composition, originally composed for a guitar and recorder duo was transcribed to be performed on the YouHero guitar, and proved very successful in this new arrangement[2].

We also gave a number of workshops for schoolchildren aged 12-17 around Sweden, centered around the YouHero instrument and contemporary music composition and performance (Fig. 4). In the workshops, the YouHero guitar was used like a normal instrument. The kids got a short introduction to the instrument, and could then play freely. Some searched for sounds similar to an electric guitar, while others started composing their own tunes. We used the instrument as an ice-breaker, to get to know them, and to start to talk about how music is made and how it can sound. Many kids have a very strong opinion about what is music and what is not music. They were definitely not without prejudice, probably because they had been taught conservative or conventional attitudes about music in school. In the concerts following the workshops, there were never any fights over who would play the YouHero guitar. When the novelty faded, it was regarded as any other instrument.

4. DISCUSSION

The YouHero sound engines are not really made for conventional tonal playing, even if this is possible. Instead, they provide exploration of complex sound spaces, thanks to the dynamic mapping algorithm and the simple but expressive control. Gesturally, it is possible to have minute, intimate

[2] `http://www.youtube.com/watch?v=Fw2I6fnXgXQ`

Figure 4: Teenagers performing on the YouHero guitar during school workshops.

control of pitch, timbre and dynamics, as well as to let go and play wild explorations. This approach is quite different from how most instruments work, and may take some time getting used to. The musicians that participated in the project were accustomed to contemporary music, but had never before encountered an instrument like this.

When evaluating an instrument, it is important to spend a lot of time acquiring skill, and to play it in many different contexts. Only then can you judge how well it works in practice, in contrast to short lab experiments, which rarely tells you something useful. To get a better picture of how the instrument works in practice, one of the main performers (Patrik Karlsson, professional guitarist and lutenist) was asked about how he felt about the instrument, after using it regularly over a couple of years in concerts and workshops. Here, we summarize his answers.

Q: How do you think the two sound engines have worked, performance-wise and timbre-wise? Any comments regarding timbral flexibility?

A: *They are different, of course, and have different conditions for use. You can always just reset all parameters and use your own effect boxes, and then you just get a normal distorted electric guitar sounds. As with all instruments, you sometimes wish for different sounds than what can be produced. They are not enormously flexible as they are.*

Q: How did the instrument turn out compared to your initial expectations?

A: *I had no idea about timbre before we started testing it. I am very much a classical musician, so I work with what I have in front of me. But before I started to tweak the sounds and the vectors, I thought it was to unreliable. Version 2 was better, with the introduction of presets.*

Q: How did the commissioned composers relate to the possibilities of the instrument, from your perspective as a musician?

A: *Only composer A really notated exact pitches and sound instructions. Maybe composer B and C practiced on the instrument, but it is not visible in the compositions. The work of composer D is the same, (s)he did not really learn how the instrument works. The music doesn't become worse when they do not know the instrument, but it is more challenging to play, technically, when they do. In the other works, the composer hands over a large part of the responsibility to find new sounds and expressions to the musicians. But we are accustomed to that...*

Q: How did you use the vector editor? Was it useful or necessary, or would a randomization function be enough?

A: *To be able to control the vectors is what makes the instrument meaningful to me. It is there I can express my personality, like my own tone when playing the lute, the guitar or any other instrument. The random function I regard more as something for those who do not know how it works, or those who do not care to practice. The vector editor works well, and I can produce sounds that I am happy*

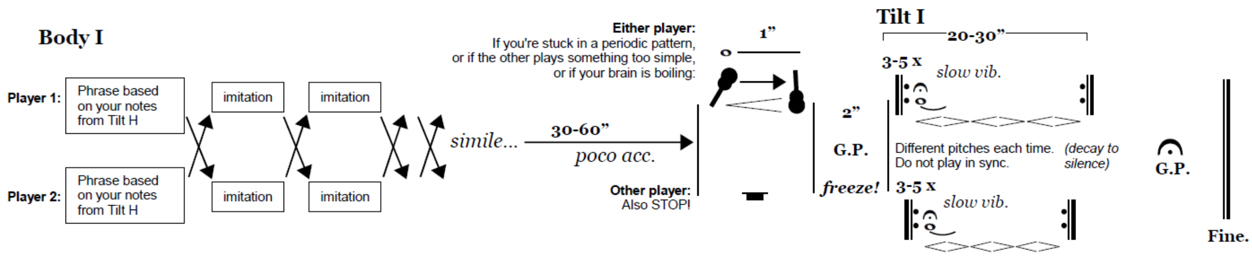

Figure 3: An excerpt from the commissioned composition *Short Term Memory* by Palle Dahlstedt. This piece uses instructions about how to interact between the players, in a kind of game piece form. Video example: `https://www.youtube.com/watch?v=bvIuDEhq-5A`.

with, save them and bring them out before each work. It also contributes to the virtuosity of the performance when you change sound and quickly create different moods.

Four essential properties of a musical instrument are: playability, repeatability/learnability, timberal control, and idiomatic playing techniques. Playability is related to variability, and there is clearly a wide range of possibilities with this instrument. Some can only be achieved using presets and on-the-fly switching, but quite complex music can also be produced with the real-time controls only. The dynamic mapping, with the origin shift mechanism which is part of the performance controls, provides means for extensive variation during a performance or during an improvisation. Still, volume is tightly coupled to timbre parameter scaling, somewhat limiting the expressive playing. Since there is no randomness involved in the performance, any playing pattern can be repeated, as long as all parameters are the same. However, the Karplus sound engine has some internal feedback paths which sometimes makes it behave in chaotic ways. As evident from the musicians involved in the project, it is possible to acquire considerable skill on the instrument. Both traditional virtuosity regarding pitch, dynamics and timing control, but also regarding navigation of the timbre space using the dynamic mapping and the more intricate process of direct vector design, essentially customizing the behavior of the whole instrument. The learning threshold is quite low, because it is easy to make cool sounds from the start. Still, there is a learning curve to attain virtuosity. This is not unlike the learning curve of a normal guitar, even though the YouHero instrument of course is much simpler.

Finally, the implementation details of any instrument gives rise to a set of playing techniques, things that are easy or interesting to play, or rewarding or challenging to learn. Some performance techniques that have emerged are:

- timbre vibrato by shaking the guitar body in specific angles
- quick alternations between sound presets
- pushing the Karplus engine into feedback and explore the highly nonlinear space of sounds there
- using the Sheperd engine with low FM strength to perform more strictly tonal material
- using the relative intervals of the neck switches to make quick tremoli on those intervals
- explore legato playing, holding the plectrum while changing pitch and timbre with the neck switches
- raising the guitar neck as a means to scale the timbre vectors, at the same time doing a theatrical gesture with rock connotations – but this can also be turned upside down, by doing the neck raise with a stone-face expression, in contrast to the cliché

4.1 Future work and conclusions

Future possibilities include an integrated sequencer system based on the GuitarHero game tablature, with both recording and playback, as well as off-line composition modes. Further sound engines would also be desirable, and should not be too difficult, since the mapping engine is generic and can be applied to any parameter set. A dream scenario would be to implement the instrument in a a game console version, but this seems unlikely, given that the GuitarHero game has fallen out of fashion, at least for the moment.

To conclude, we successfully developed a concert instrument from a computer game guitar controller, which in spite of its simplicity allows for a wide range of playing techniques and sophisticated timberal control. It is not so suitable for traditional music, but lends itself well to improvisation and experimental explorations. Some experimental features turned out to be essential for professional use, such as the vector editor, while the sound engines were perceived as difficult by some. The initial learning threshold is low, but it is still possible to build up considerable skill on the instrument. It has been used in a number of school workshops to inspire kids to create experimental music, as well as a concert instrument for a series of commissioned compositions from professional composers. We did attain our goals, learned a lot on the way. And we had fun doing it.

5. REFERENCES

[1] C. Cunningham. The guitar zeros @ maker faire. Make: online. http://makezine.com/2008/04/01/the-guitar-zeros-maker-fa/.

[2] P. Dahlstedt. *Sounds Unheard of: Evolutionary algorithms as creative tools for the contemporary composer*. PhD thesis, Chalmers University of Technology, 2004.

[3] P. Dahlstedt. Dynamic mapping strategies for expressive synthesis performance and improvisation. In *LNCS 5493*, pages 227–242, 2009.

[4] C. Goudeseune. Interpolated mappings for musical instruments. *Organised Sound*, 7(2):85–96, 2002.

[5] M. Luhtala, T. Kymäläinen, and J. Plomp. Designing a music performance space for persons with intellectual learning disabilities. In *NIME*, 2011.

[6] A. McPherson and Y. Kim. The problem of the second performer: building a community around an augmented piano. *Computer Music Journal*, 36 (4), 2012.

[7] C. Reiley. Air guitar hero. *Make*, 29:44–51, 2012.