Kiroll: A Gaze-Based Instrument for Quadriplegic Musicians Based on the Context-Switching Paradigm

Nicola Davanzo
Luca Valente
Luca Andrea Ludovico
Federico Avanzini
nicola.davanzo@unimi.it
Laboratory of Music Informatics, University of Milan
Milan, Italy

ABSTRACT

In recent years, Accessible Digital Musical Instruments (ADMIs) designed for motor-impaired individuals that incorporate gaze-tracking technologies have become more prevalent. To ensure a reliable user experience and minimize delays between actions and sound production, interaction methods must be carefully studied. This paper presents *Kiroll*, an affordable and open-source software ADMI specifically designed for quadriplegic users. *Kiroll* can be played by motor-impaired users through eye gaze for note selection and breath for sound control. The interface features the *infinite keyboards context-switching* interaction method, which exploits the smooth-pursuit capabilities of human eyes to provide an indefinitely scrolling layout so as to resolve the Midas Touch issue typical of gaze-based interaction. This paper outlines *Kiroll*'s interaction paradigm, features, implementation processes, and design approach.

CCS CONCEPTS

• **Applied computing → Sound and music computing**; • **Human-centered computing** → *Gestural input*; **Accessibility technologies**.

KEYWORDS

digital musical instruments, accessibility, eye tracking

ACM Reference Format:
Nicola Davanzo, Luca Valente, Luca Andrea Ludovico, and Federico Avanzini. 2018. Kiroll: A Gaze-Based Instrument for Quadriplegic Musicians Based on the Context-Switching Paradigm. In *Proceedings of AudioMostly '23.* ACM, New York, NY, USA, 4 pages. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

Music is an activity that is easily available to many people, with cognitive and non-musical benefits. For instance, learning to play an instrument can enhance spatial reasoning skills in children [4], and engagement with music can enrich non-musical abilities across different age groups [12]. Additionally, musical participation promotes self-esteem and personal growth across development, such as during school performances [3]. However, due to the hand-based nature of most instruments, quadriplegics are often excluded from engaging in conventional musical activities. Quadriplegia is the paralysis of all four limbs and the torso, caused by conditions such as spinal cord injuries, multiple sclerosis, amyotrophic lateral sclerosis, and transverse myelitis [22]. To address this gap, Accessible Digital Musical Instruments (ADMIs) that primarily employ eye tracking technology have been suggested in the literature. However, delay-free interactions are necessary to prevent complications such as Delayed Audio Feedbacks (DAF) that hinder proper musical performance [18]. These techniques can provide a benchmark for eye tracking interaction across general-purpose applications.

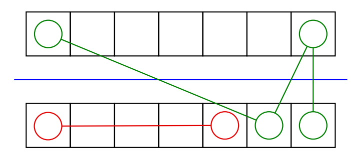

Figure 1: The context-switching paradigm proposed in [17]. Alternating gaze between the two keyboards avoids crossing intermediate keys (green trace), which in turn would happen if only a keyboard was present (red trace).

In this regard, *Kiroll*[1] is a software ADMI controlled by gaze tracking for note selection and breath pressure for sound intensity and dynamics. The interface introduces a novel interaction paradigm called *infinite keyboards context-switching* to bypass the Midas Touch issue, which represents the main contribution of this work. The instrument resembles a melodica capable to play one note at a time, where eyes are used instead of fingers, and keys are moving on the computer screen. *Kiroll* and its source code are available in a GitHub repository under GNU GPL-v3 Free Open-Source License.[2] A link to an explicative demo video of the instrument is provided in the readme file.

2 BACKGROUND

Davanzo and Avanzini [6] categorized interaction options for quadriplegic individuals into four groups: *eyes* (including gaze pointing,

[1]The name means "Keyboards infinite roll".
[2]*Kiroll*'s GitHub repository: https://github.com/LIMUNIMI/Kiroll

AudioMostly '23, 30 Aug - 1 Sep 2023, Edinburgh Napier University, UK

ACM ISBN 979-8-4007-0818-3…$15.00
https://doi.org/XXXXXXX.XXXXXXX

eye movements, blinking, and eyebrow movements), *mouth* (including voice, whistling, breathing, mouth and lip movements, tongue, and teeth), *head* (incorporating rotational coordinates and neck tension), and *brain* (encompassing mental state, attention, and motor imagery). They coined the term *HeaDMIs* to refer to a group of instruments like *Kiroll*, designed for quadriplegics to simulate the interaction mechanics of traditional acoustic instruments. These instruments can be controlled through skill-based movements or smooth automated actions that do not require focused attention but rely on continuous environmental feedback for high-level integration and temporal precision [6]. Gaze-based methods have been widely used in both accessible and general-purpose musical interfaces. This approach has a strong track record in the literature, with early sonification experiments such as *Intuitive Ocusonics* [19] and *Oculog* [15], as well as more comprehensive platforms like *Eye Play the Piano* [11], which offers users the opportunity to play a physical piano through actuators while interacting with a VR interface. Various articles in the literature enumerate the different types of movements that the human eye can execute [16]. Rapid and jerky movements (*saccades*) make the gaze point shift from one discrete point to another, interspersed with almost immobile positions (*fixations*). However, the eye needs a target to focus on to perform smooth movements (*smooth pursuits*), which results in a peculiar motion that has been widely studied from a neurological and physiological point of view [2]. Mathematical models have also been provided [20]. Smooth pursuit-based paradigms were also tested for calibration-less gaze interaction in smart watches [10] and VR headsets [14]. As detailed in Sec. 3.1, these movements are of primary importance in the operation of *Kiroll*.

Over time, advanced interfaces have been developed to address various gaze interaction issues. Davanzo and Avanzini [7] provide an overview, as well as several techniques and cues to address them. A major challenge is posed by the Midas Touch problem [16]. This challenge arises when using visual pointing via eye tracking to control an interface that has multiple gaze-sensitive elements, which can be activated by involuntary or exploratory gaze movements. Moreover, even if the user learns to limit them, modern eye trackers still detect several intermediate points during saccades. This detection causes all the buttons present in the gaze path to activate and leads to unintended actions and effects, as demonstrated in Fig. 1 (red trace). Designers of advanced interfaces have developed various strategies to address this key challenge.

On top of this, *HeaDMI* interaction demands high temporal accuracy. In the context of non-time-critical interaction, typical solutions to tackle the Midas Touch involve the implementation of dwell-time (delayed selection) or fixation-discrimination filters [7]. Nonetheless, such approaches may lead to DAF on the user's physical actions, which are known to negatively impact the quality of musical performance [18]. Wessel and Wright recommend that the delay on audible system reactions should not exceed 10 ms [24]. ADMIs that avoid such filtering have shown that the absence of DAF can improve the temporal accuracy of performance [9, 17]. An alternative solution to the Midas Touch problem was provided by *Lumiselo* [1] and later introduced into *Netytar* [9] and *Netychords* [8]. This approach, previously called *hybrid interaction* [7], involves splitting the interaction into two channels: selection through gaze and activation through other gestures (e.g., breath pressure or head rotation). As detailed in Sec. 3.2, *Kiroll* builds on this mapping solution by proposing the use of breath, detected through a sensor, as a method for controlling the onset of notes and sound intensity. Finally, appropriate key layouts let the user move from a note to another by limiting the number of intermediate keys the gaze must pass through or even eliminating them. Such solutions include pie-shaped layouts, as the one employed in *The EyeHarp* [23], and particular isomorphic layouts, as proposed in *Netytar*.

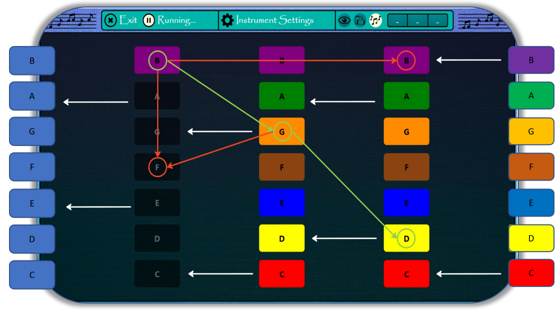

Figure 2: *Kiroll*'s interface and its infinite keyboard context-switching paradigm. Green circles and lines depict normal gaze traces during the performance, red ones forbidden gaze movements, and arrows the auto-scrolling motion.

Morimoto *et al.* introduced an innovative layout known as the *context-switching paradigm* (Fig. 1), which was implemented in the *Eyejam* instrument [17]. This approach involves dividing the interface into two identical keyboards, arranged in parallel, and separated by a space called the *bridge*. The bridge can either be empty or contain additional information for the user, such as a music score. To play a new note, the user needs to shift the gaze across the bridge. By alternating between the two keyboards, the user's gaze can thus avoid crossing intermediate keys. The authors introduced metrics to measure errors in note sequences and temporal cadence, including rhythm and phase. They conducted an experiment comparing two versions of the interface, one based on dwell-time and the other based on the context-switching paradigm. In tests performed at 70 BPM, the context-switching-based interface demonstrated a significant reduction in average phase (from 170 ms to 145 ms) and rhythm (from 170 ms to 100 ms) errors and fewer errors in notes sequences.

3 IMPLEMENTATION

Building on Morimoto et al.'s context switching paradigm [17], *Kiroll*'s objective is to propose a similar approach with the addition of smooth automatic scrolling through an infinite number of keyboards, exploiting the smooth pursuit capabilities of the human eye. Eye movements while playing are similar to reading music on a staff, thus paving the way for eye-based music composition tools.

3.1 Interface

Differently from *Eyejam*, *Kiroll*'s layout (Fig. 2) features an infinite number of parallel vertically-arranged keyboards. Each colored key on the keyboard represents a specific note. The interface works as follows: **(a)** The musicians select a key on the leftmost keyboard by

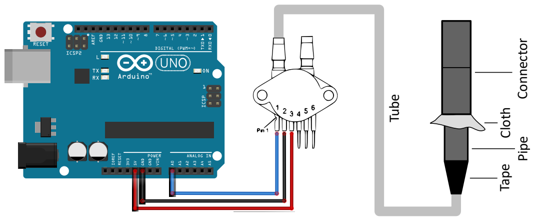

Figure 3: The building scheme for *NITHbreathSensor*.

placing their gaze on it; **(b)** Then they blow into the breath sensor to raise a MIDI Note-On event (full velocity). Channel Pressure will follow breath pressure in order to control sound intensity. All the other keys on that keyboard will be dimmed to indicate they can no longer be played; **(c)** Their gaze will then move (through a saccade) to a key on the next keyboard to the right, corresponding to the next desired note, as depicted by the green lines in Fig. 2. If the musicians are still blowing into the sensor, that note will be immediately played. This system allows to play notes legato without the need to cross intermediate keys and avoids the Midas Touch issue (Sec. 2), as shown in the figure.

Through a system named *Autoscrolling*, the interface will automatically scroll to the left in response to the musician's gaze. As a result, the leftmost keyboard visible on the screen will move off it, and the second keyboard - smoothly followed by the musician's gaze - will become the new first keyboard on the left. *Autoscrolling* has been previously suggested in [7] and implemented in the *HeaDMIs Netytar* and *Netychords* [8, 9]. The sliding velocity v_a is exponentially proportional to the distance between the left side of the interface x_0 and the current position x_g of the gaze on the horizontal axis: $v_a = [(x_g - x_0) \cdot a]^b$, where a and b are two adjustable constants. Furthermore, to avoid sudden spikes in keyboard motion, the gaze position x_g which inputs to the autoscrolling system at instant t is filtered through an exponentially-decaying moving-average filter, in the form: $x_g(t) = x_g(t-1) \cdot \alpha + x_i(t) \cdot (1-\alpha)$, where α is an arbitrary constant in the interval $(0; 1)$, and x_i is the raw sample from the eye tracker. The resulting gaze motion while playing is akin to observing static objects while looking out of a train window.

Kiroll also incorporates design cues from [7], including the use of highly contrasting colors that leverage the human peripheral vision's sensitivity to colors. Feedback is provided without on-screen cursors to prevent distractions, while visual references at the center of the key provide gaze hooks.

Users can adjust various settings, through the option menu or through blinking/winking gestures: musical scale, octave, distance between keyboards, keyboards orientation. To mitigate the effect of noisy eye tracking signal, buttons are thoughtfully spaced apart, and the last gazed item is highlighted in yellow, even if the gaze signal is out of its area. To activate the selected item, users can exhale briefly or perform a prolonged blink using both eyes. This allows for self-sufficient navigation for quadriplegic users without external assistance.

3.2 Hardware and Software

Kiroll detects breath pressure through *NITHbreathSensor*, a deliberately simple open-source breath sensor built using Arduino, a *MXP5010DP* 10 KPa air pressure sensor and DIY materials (Fig. 3). It belongs to *NITHsensors*[3], an open-source collection of sensor peripherals we developed to provide motor-impaired users with easy and affordable access to sensing technology. This is achieved through several key concepts. Firstly, they are easy to build through DIY practices and simple instructions that require no specific electronics or manufacturing skills. Secondly, the use of open-source microprocessors allows for easy replication of the peripherals, keeping costs low. Finally, the construction of peripherals, as well as control software, libraries, and APIs, are published under free licenses, thus enhancing hardware reproducibility and scientific transparency.

Kiroll was developed using C# and the WPF framework to define the GUI. The instrument was implemented using the not yet published development library *NITHdmis* [5][4], which allows for the rapid and modular prototyping of *HeaDMI*s. The library uses a "rack" metaphor to organize the code into different modules, contained in a single rack in a manner similar to organizing sound modules (e.g., synthesizers, MIDI outputs, control modules, etc.). *Kiroll*'s music mapping logic is contained in the `KirollDMIBox` module, while the `MIDIModule` is dedicated to sending MIDI messages. A noteworthy module is the `NithModule`, which is responsible for receiving and interpreting input from the *NITHbreathSensor* connected to the USB port. This module provides a strategy to independently support various sensors in the NITHsensors collection and easily introduce custom mappings to arbitrary peripherals. Furthermore, the incoming input strings from the sensor at each sampling cycle, received and interpreted by `NithModule`, use a standard format:

`sensorname − version|opcode|arg1 = val&arg2 = val&...`

The arguments (*arg1*, *arg2*, etc.) specify the input data type (e.g. breath pressure). Flexibility is a key feature of the `NithModule` providing the abstraction of the connected peripheral model, ensuring the required arguments (e.g., pressure for *Kiroll*) are present within the input string, thus allowing easy input data management, as well as customization or implementation of new peripheral models.

Gaze detection in *Kiroll* is achieved through a low-cost non-invasive Near-Infrared camera-based commercial eye tracker from Tobii. The software is compatible with various Tobii eye trackers, such as *Tobii Eye Tracker 5*.[5] As there is approximately 1° of signal noise with respect to the visual angle when using this device [13], the interface requires larger buttons and visual elements.

4 DISCUSSION AND FUTURE PERSPECTIVES

It is worth noting that this layout, featuring infinite keyboards that leverage the smooth pursuit capabilities of the human eye, would arguably be ineffective in a fingered instrument, as users would have to track the moving keyboards with their fingers.

Like *Eyejam*, *Kiroll* features multiple parallel keyboards separated by a bridge space, but there are significant differences: **(a)** In *Kiroll*, gaze movement progresses single-handedly from left to

[3]NITHsensors instructions, details, and source codes: https://github.com/LIMUNIMI/NITHsensors
[4]NITHdmis on GitHub: https://github.com/LIMUNIMI/NITHdmis
[5]Tobii's Eye Tracker 5: https://gaming.tobii.com/product/eye-tracker-5/

right, whereas *Eyejam* involves continuous bouncing between the keyboards; **(b)** *Kiroll*'s keyboards are in motion and the eye follows through smooth pursuit, unlike *Eyejam*'s stationary keyboards; **(c)** *Kiroll*'s breath sensor can control the sound intensity and notes onsets, potentially mitigating Midas Touch, while *Eyejam* lacks methods for intensity control.

We experimented with *Kiroll*'s paradigm due to several factors. Firstly, its unique movement direction may be better suited for musical sequences featuring an odd number of notes, resulting in a more symmetrical experience. Secondly, the vertical note arrangement follows the natural sequence of pitches in the *SMARC* effect [21]. Thirdly, the horizontal keys sequence can represent the passage of time, much like reading a staff or piano roll, potentially allowing users to learn *Kiroll* quickly by building on their existing music reading skills. We may explore incorporating a musical staff into the interface as we continue to refine *Kiroll*. The resemblance to a musical staff makes *Kiroll* a candidate for future development as a platform for accessible music composition software. Additionally, simple and intuitive gamification solutions could be introduced, as the horizontal arrangement of the keyboards resembles the popular game *Guitar Hero*.[6]

A main limitation of *Kiroll*'s interface lies in the limited number of notes that can be displayed onscreen, resulting from the inability to utilize both dimensions of the screen to represent more keys, as is the case with *Netytar* and *The Eyeharp* [9, 23]. Activating vertical autoscrolling allows the user to view additional hidden keys; however, this solution does not permit octave intervals to be played. Moreover, tactile feedback systems could be introduced to enhance the experience.

The usability of *Kiroll* has not been evaluated so far, in particular with impaired users. Concerning future test activities, investigations will explore the following research questions: **(a)** Are the vertical scrolling and smooth pursuit movements intuitive and natural for all users? **(b)** How does the instrument perform in terms of timing errors and incorrect notes in comparison to context-switching interfaces such as *Eyejam*? **(c)** Which aspects, such as smooth pursuit characteristics or saccades per second, may be a bottleneck for notes performance speed? To answer these questions and gain insights into user perspectives, Kiroll must undergo an empirical evaluation. Conducting a comparative experiment between *Kiroll* and *Eyejam* could be beneficial. The experiment could measure the average pairwise difference in subject performance while using both systems, using the metrics introduced by Morimoto *et al.* to assess error rate and timing inaccuracies. Additionally, explorative case studies through interviews, questionnaires, and think-aloud could help to gain further understanding.

REFERENCES

[1] Sam Bailey, Adam Scott, Harry Wright, Ian M. Symonds, and Kia Ng. 2010. Eye.Breathe.Music: Creating Music through Minimal Movement. In *Proc. Conf. Electronic Visualisation and the Arts (EVA 2010)*. ScienceOpen, London, UK, 254–258.

[2] G. R. Barnes. 2008. Cognitive Processes Involved in Smooth Pursuit Eye Movements. *Brain and Cognition* 68, 3 (Dec. 2008), 309–326. https://doi.org/10.1016/j.bandc.2008.08.020

[3] Eugenia Costa-Giomi. 2004. Effects of Three Years of Piano Instruction on Children's Academic Achievement, School Performance and Self-Esteem. *Psychology of Music* 32, 2 (April 2004), 139–152. https://doi.org/10.1177/0305735604041491

[4] Rudi Črnčec, Sarah J. Wilson, and Margot Prior. 2006. The Cognitive and Academic Benefits of Music to Children: Facts and Fiction. *Educational Psychology* 26, 4 (Aug. 2006), 579–594. https://doi.org/10.1080/01443410500342542

[5] Nicola Davanzo. 2022. *Accessible Digital Musical Instruments for Quadriplegic Musicians*. Ph. D. Dissertation. Università degli Studi di Milano, Milano.

[6] Nicola Davanzo and Federico Avanzini. 2020. Hands-Free Accessible Digital Musical Instruments: Conceptual Framework, Challenges, and Perspectives. *IEEE Access* 8 (2020), 163975–163995. https://doi.org/10.1109/ACCESS.2020.3019978

[7] Nicola Davanzo and Federico Avanzini. 2022. Design Concepts for Gaze-Based Digital Musical Instruments. In *Proceedings of the 2022 Sound and Music Computing Conference*. Zenodo, Saint-Etiénne, France, 477–483.

[8] Nicola Davanzo, Matteo De Filippis, and Federico Avanzini. 2021. Netychords: An Accessible Digital Musical Instrument for Playing Chords Using Gaze and Head Movements. In *In Proc. '21 Int. Conf. on Computer- Human Interaction Research and Applications (CHIRA '21)*. SciTePress, Online conf., 8.

[9] Nicola Davanzo, Piercarlo Dondi, Mauro Mosconi, and Marco Porta. 2018. Playing Music with the Eyes through an Isomorphic Interface. In *Proc. of the Workshop on Communication by Gaze Interaction*. ACM Press, Warsaw, Poland, 1–5. https://doi.org/10.1145/3206343.3206350

[10] Augusto Esteves, Eduardo Velloso, Andreas Bulling, and Hans Gellersen. 2015. Orbits: Gaze Interaction for Smart Watches Using Smooth Pursuit Eye Movements. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology (UIST '15)*. Association for Computing Machinery, New York, NY, USA, 457–466. https://doi.org/10.1145/2807442.2807499

[11] Fove Inc. 2020. Eye Play the Piano. http://eyeplaythepiano.com/en/.

[12] Henkjan Honing, Olivia Ladinig, Gábor P. Háden, and István Winkler. 2009. Is Beat Induction Innate or Learned? Probing Emergent Meter Perception in Adults and Newborns Using Event-Related Brain Potentials. *Annals of the New York Academy of Sciences* 1169, 1 (2009), 93–96. https://doi.org/10.1111/j.1749-6632.2009.04761.x

[13] Andrew Housholder, Jonathan Reaban, Aira Peregrino, Georgia Votta, and Tauheed Khan Mohd. 2022. Evaluating Accuracy of the Tobii Eye Tracker 5. In *Intelligent Human Computer Interaction (Lecture Notes in Computer Science)*, Jong-Hoon Kim, Madhusudan Singh, Javed Khan, Uma Shanker Tiwary, Marigankar Sur, and Dhananjay Singh (Eds.). Springer International Publishing, Cham, 379–390. https://doi.org/10.1007/978-3-030-98404-5_36

[14] Mohamed Khamis, Carl Oechsner, Florian Alt, and Andreas Bulling. 2018. VRpursuits: Interaction in Virtual Reality Using Smooth Pursuit Eye Movements. In *Proceedings of the 2018 International Conference on Advanced Visual Interfaces (AVI '18)*. Association for Computing Machinery, New York, NY, USA, 1–8. https://doi.org/10.1145/3206505.3206522

[15] Juno Kim, Greg Schiemer, and Terumi Narushima. 2007. Oculog: Playing with Eye Movements. In *Proceedings of the 7th International Conference on New Interfaces for Musical Expression - NIME '07*. ACM Press, New York, New York, 50. https://doi.org/10.1145/1279740.1279747

[16] Päivi Majaranta and Andreas Bulling. 2014. Eye Tracking and Eye-Based Human–Computer Interaction. In *Advances in Physiological Computing*, Stephen H. Fairclough and Kiel Gilleade (Eds.). Springer, London, 39–65. https://doi.org/10.1007/978-1-4471-6392-3_3

[17] Carlos H. Morimoto, Antonio Diaz-Tula, José A. T. Leyva, and Carlos E. L. Elmadjian. 2015. Eyejam: A Gaze-Controlled Musical Interface. In *Proceedings of the 14th Brazilian Symposium on Human Factors in Computing Systems (IHC '15)*. ACM, Salvador, Brazil, 37:1–37:9. https://doi.org/10.1145/3148456.3148493

[18] P. Pfordresher and C. Palmer. 2002. Effects of Delayed Auditory Feedback on Timing of Music Performance. *Psychological Research* 66, 1 (Feb. 2002), 71–79. https://doi.org/10.1007/s004260100075

[19] Andrea Polli. 1999. Active Vision: Controlling Sound with Eye Movements. *Leonardo* 32, 5 (Oct. 1999), 405–411. https://doi.org/10.1162/002409499553479

[20] D. A. Robinson, J. L. Gordon, and S. E. Gordon. 1986. A Model of the Smooth Pursuit Eye Movement System. *Biological Cybernetics* 55, 1 (Oct. 1986), 43–57. https://doi.org/10.1007/BF00363977

[21] Elena Rusconi, Bonnie Kwan, Bruno L. Giordano, Carlo Umiltà, and Brian Butterworth. 2006. Spatial Representation of Pitch Height: The SMARC Effect. *Cognition* 99, 2 (March 2006), 113–129. https://doi.org/10.1016/j.cognition.2005.01.004

[22] Andrew Sears, Mark Young, and Jinjuan Feng. 2008. Physical Disabilities and Computing Technologies: An Analysis of Impairments. In *The Human-Computer Interaction Handbook* (2 ed.), Andrew Sears and Julie A. Jacko (Eds.). CRC Press, United States, Chapter 42, 829–852.

[23] Zacharias Vamvakousis and Rafael Ramirez. 2016. The EyeHarp: A Gaze-Controlled Digital Musical Instrument. *Frontiers in Psychology* 7 (2016), article 906. https://doi.org/10.3389/fpsyg.2016.00906

[24] David Wessel and Matthew Wright. 2017. Problems and Prospects for Intimate Musical Control of Computers. In *A NIME Reader: Fifteen Years of New Interfaces for Musical Expression*, Alexander Refsum Jensenius and Michael J. Lyons (Eds.). Springer International Publishing, Cham, 15–27. https://doi.org/10.1007/978-3-319-47214-0_2

[6]Guitar Hero on Wikipedia: https://en.wikipedia.org/wiki/Guitar_Hero