A preliminary study of Augmented Musical Instruments for Study (AMIS) using research through design.

Eduardo Santos Silva [1 2 *], Giordano Ribeiro Eulalio Cabral [1 2], Rodrigo Mendes de Carvalho Pereira [2]

[1] Centro de Informática – Universidade Federal de Pernambuco
Av. Prof. Moraes Rego, 1235 - Cidade Universitária, Recife - PE, 50670-901

[2] Daccord Music Software
R. Vig. Tenório, 194 - Sala 401 - Recife, PE, 50030-010

ess@cin.ufpe.br, grec@cin.ufpe.br, rodrigo@daccord.com.br

Abstract

The emergence of Digital Musical Instruments (DMIs) in the computer music field has been providing new means of interaction with music performances. Particularly, with the advancements in the Internet of Things (IoT) area, there has been an increase in augmented musical instruments containing LEDs within their own bodies to support music learning. These instruments can help musicians by having the capabilities to display musical notes, chords and scales. This research uses research through design to present a preliminary study of some of the challenges related to the design of these instruments and attempts to provide some insights associated with their usability and development, particularly related to the development of an augmented acoustic guitar, the VioLED, developed alongside the company Daccord Music. The system provides three modes of operation: song mode, solo/improvising mode and animation mode. The preliminary results present challenges and obstacles pertinent to usability and development of these instruments identified in different iterations of design. These challenges are associated with areas such hardware-software co-design, usability, latency/jitter, energy consumption, etc.

1. Introduction

In recent years, the computer music area has witnessed the emergence of several technological advancements that allow new means of audio and musical interactions. Applications that involve from new ways of song playback [1] to new Digital Musical Instruments [2] and smart instruments [3]. Particularly, in the field of gestural interfaces, the technological advancements enable the development of new musical instruments augmented with sensors, actuators and microcontrollers to provide new features to the user, also known as hyperinstruments, to extend their sound capabilities [2].

1.1 “The perfect storm”

In the last few years, the IoT research area has gained attention with the increasing popularity of mobile consumer electronics such as smartphones, tables, wearable devices, etc. The increasing number of devices connected to the internet [4] and the increasing accessibility of broadband mobile internet connection, microcontrollers, e.g., Arduino, Raspberry Pi, wireless communication technologies, e.g., Bluetooth, Xbee, NFC, as well as sensors and actuators, produce a scenario that can stimulate the development of new electronic projects. Simultaneously, the growth of music streaming services, e.g., Spotify, Apple Music, Deezer, Tidal, etc. provide expanding content to new music applications that vary from IoT Playback devices, e.g., Aumeo, audio customization, setups for collaborative creation, devices for artist-audience interaction, smart instruments, etc. [3].

1.2 Augmented Musical Instruments for Study (AMIS)

In the context of musical learning, there has been an appearance of instruments a particular set of augmented instruments, referred in this paper as Augmented Musical Instruments for Study (AMIS), which use light-emitting diodes (LEDs), projection and other means to display musical content directly on the instrument’s body as means to help beginner musicians.

Several researchers and commercial products have focused on augmenting musical instruments to support music learning [5, 6, 7, 8, 9, 10, 11, 12]. These systems can display musical notes, chords, scales and the vast musical content and body of technique, developed across generations, by using different approaches. A few of the systems use highly intrusive approach by implanting LEDs and Print Circuit Boards within the instruments manufacturing process [5, 8, 9, 10]. Other systems still utilize LEDs, however, exploit a non-intrusive approach by creating a “sleeve” that can be inserted over the instrument [6]. Other researchers use projection systems to display the musical content over the musical instrument. These systems [10, 11] use software to convey notes and chords to the user. Furthermore, there are approaches that use the Head-Mounted Displays to support music learning, e.g., Music-Everywhere [12].

1.3 Applications and services

Different applications and services can provide musical content and tools to the learning process. Musicians often use tablature and chord charts services

* Supported by FACEPE

such as Ultimate Guitar, Cifra Club, Guitar Pro, Songsterr, etc. in the learning process. These provide musical content with different music notations (sheet music, tablatures, chord charts). Video sharing websites, e.g., Youtube, have become more popular among musicians as a source of tutorials created by the users. Furthermore, software such as Yousician, Synthesia, Chordify, Yalp, etc. provide interactive content to support music learning also emerge with different approaches incorporating social network and gamification elements.

2. Challenges in the design process of the AMIS

These AMIS exist in a very particular design context that present obstacles in different areas: HCI, Real-Time Applications, Hardware-Software co-design and music learning.

2.1 Challenges in HCI

Some researches debate that the role of interaction design is to provide usability to the design process by making systems safe, reliable, easy and pleasant to use at the same time that they are functional [13]. The challenge is to provide usability according to different aspects that can affect the system, such as organizational aspects, environment, safety and security, user engagement, comfort, task related aspects, user interface, among others. Further details about the criteria, the different stakeholders involved in system's usability is present in the literature [13].

The research of Dix et. al. [14] and Nielsen [15] provide insight on different criteria to improve usability. However, the challenge remains in identify the trade-offs between each system's priority features and their effect on the overall usability. As an example, the vast number of features in Adobe's Photoshop may be very helpful to users, but the amount of content can increase the intricacy of its user interface and might cause an overwhelming sensation at first.

Moreover, the evaluation of these criteria is somewhat complex, involving qualitative research with specialists and end-users. Some researchers discuss that for mobile applications, interactions design has some limitations related to different devices' components, screen size and navigation structures [16]. Particularly in the case of the AMIS, the systems and services for music learning have real time synchronized interactions, a variety of different complexity content, and interaction elements that can influence the usability and how these elements can affect the learning experience.

2.2 Challenges in hardware/software co-design (HW/SW co-design)

Predominant in development of integrated circuit systems, especially in embedded systems, research in the literature involves analyze how to address system requirements through hardware and software synergy [17].

Several challenges in HW/SW co-design refer to management and tools for different teams working in parallel, specific tests, etc. For real time applications, requirements in performance can be greater than control systems. However, control systems deserve greater attention in reliability, safety/security, etc. Some authors highlight three aspects during the design process: modeling, validation and implementation [17]. In IoT projects, these challenges emerge when designing new systems, since most specific and proprietary applications need custom hardware and software. Moreover, interactive systems connected to mobile devices often need to assess which features are better implemented in hardware and the ones are more suited for software. This is a particular challenging design problem when dealing with time-to-market [18].

Furthermore, HW/SW co-design can be a challenge in the prototyping phase. For systems that lack flexibility regarding changes and improvements in hardware, it can hinder both exploration within the design space as well as the freedom to meet new software requirements [18], therefore, further delaying the design process.

2.3 Challenges in real time applications

Real time systems depend not only from the information and system behavior, but also from how long it takes to the information to arrive or the functionality to occur. These systems highly demand from precision and reliability on the information and system's response, since any errors or delay can have fatal safety and security issues or completely interfere in the user's experiences.

DMIs and AMIS also have the real time requirements since "time is central feature in music" [19]. Therefore, errors related to package loss from the communications module or latency can hinder the experience by showing incorrect notes at incorrect times. Latency and jitter are common challenges when designing DMIs and therefore AMIS [20].

The prototyping phase of AMIS can also be challenging because deals with tasks that increase in complexity. Similarly to the game development process, it becomes challenging to envision all the functionalities and elements involved in the experience by paper prototypes or models [22].

2.4 Challenges in music learning

As discussed in previous sections, both hardware and services associated present challenges in different areas during the design process. However, these services aim to support music learning therefore on must consider aspects related to informal/formal learning, types of music notation, learning new abilities and their learning curve.

Learning a music instrument involves acquiring a whole body of technique developed across generations [23]. Both cognitive abilities to recognize and understand music notations and structures [24], but also motor skills to apply these music concepts into the instrument [19, 21, 24].

Some authors discuss different types of musical learning processes. While formal learning focus on understanding music notations, and the body of technique of that instrument [23], training virtuosos [25], informal learners use video lessons, friends, family, simpler music notation (tabs and chord charts), popular repertoire and focus on “play music” instead of “how to play and compose music” [26].

The graphical notation also can influence the understanding of the musical concepts. While sheet music provides more complex and deeper understanding of all aspects related to notes, music structures, it may become cumbersome for beginner musicians to learn at the same time they need to acquire motor skills to apply these techniques. In contrast, tablatures and chord charts provide a simpler notation to understand music to the detriment of some musical aspects, such as time of execution of each note and chord.

Some research discuss that the learning curve of musical instruments relates to the time taken to achieve certain level of skill to master the instrument or its body of technique [27]. Some authors argue that it can take up to ten years to achieve this point [19]. Other research discuss that the instrument needs to provide easy access to understand and achieve results at the same time they provide enough challenge to master [28].

3. Method

This section discusses the method and design tools used in this research.

Zimmerman et. al. discuss that the research through design method emerged as tools for designers to handle complex systems and “wicked problems”, i.e., problems that are too difficult to model because of conflict between its stakeholders and aspects [29]. Other authors use the term “design-oriented research” to define the use of design methods to produce academic research and knowledge [30] by developing, testing and evaluating prototypes. Prototyping is an essential step when de designing new systems, as each iteration containing tests and evaluation of the prototype can reveal flaws, features that need to be removed and essential requirements, reducing workload in non-essential features and guiding the design process.

The method used in this paper follow an iterative process containing three steps: learn, build and measure (Figure 1).

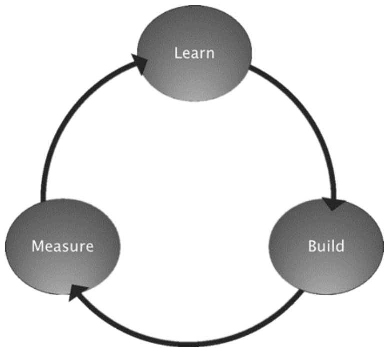

Figure 1: Design method

The learning phase involve understanding the context, collecting interviews and reports about the about the subject, studying market trends and analyze related work in the literature. The build phase involved using design tools to produce a prototype containing basic essential features for evaluation in the measure phase. This step is essential to identify the system’s requirements. The methods applied were Brainwriting 6-3-5 and a modification of Crawford Slip. The modification on the Crawford Slip method was integrating sketches in idea elaboration so the participants could elaborate how the user interaction would be. Furthermore, prioritizing ideas and specifying requirements based on innovation, target audience, market strategy, music learning and technological aspects. In the measure stage, the internal team and potential users test the prototype to evaluate which features need to be discarded and which features advance to the next iteration.

3.1 Learn phase

After reviewing literature research, the next step was conducting interviews to identify key aspects as well as gather the design team to perform the brainwriting and Crawford Slip methods. Because of the ongoing COVID-19 pandemic, the method for conducting interviews was through Skype and Zoom calls with 27 subjects between March and May 2020.

The subjects vary between beginners, professional and aspiring musicians, as well as people who previously tried to learn guitar. The interviews tried to assess their experiences with music learning, discussions about learning methods and possible features that were lacking when they learn (or attempted to learn) the instrument.

Analyzing the interviews, the subjects discussed key aspects: learning chord positions, feedback, improve technique, freedom to learn in mobile devices and finding song within their level that they wanted to learn. Some interviewers described that their learning experience with professional tutors or schools were mainly formal learning and that discourage them to pursue further learning or that they “endure” the process. One subject reported that they wanted to “first learn how to play, theory comes later”.

When questioned about how do they thought music learning was going to be in 15-20 years, most described learning through mobile devices, but still respecting the tutor’s role. Three subjects even reported the possibility of the emergence of “devices connected to the instrument to see chord charts”, “an augmented guitar that shows chord charts” and “a guitar with some kind of holographic fretboard” to “see the chords in the fretboard”.

3.2 Build phase

Before starting building the prototype, a team containing six individuals between 20 and 40 years old from different departments (three from management department, two from development and one from marketing) participate in the brainwriting 6-3-5 and variation of Crawford Slip methods.

The results identified five key features: legibility, integration with vast song databases, social network, gamification elements and simple design. In this context, legibility relates not only to the display format on screen (size, font, etc.), but also to the understanding of the content itself. That way, the music notation chosen at first was the chord charts notation. Integration to vast song databases were related to associating the musical learning with music streaming services, which led to synchronizing music learning with song playback. Additionally, Elements in social network and gamification can improve user's motivation, which could help users that previously tried to learn the instrument and lost interest or aspiring musicians who look for an informal learning style.

4. The VioLED prototype

The VioLED (Figure 2) is an acoustic guitar containing a PCB board with LEDs behind the instrument's fretboard, developed in a partnership between the companies Daccord Music and Batebit Artesania Digital. The company Batebit developed the hardware and performed all its changes across versions, as well as the prototype's version 0.1, working alongside a luthier to insert the electronics into the instrument.

The prototype aims to benefit from IoT technologies to support music learning. The development of the prototype also resulted in two patents related to the instrument itself (BR1020170083071) and its communication protocol (BR1020170082962).

Figure 2: VioLED.

All versions of the prototype use a screwd-in PCB board containing 78 WS2812B RGB individually addressed LEDs. Each LED represents one note in the fretboard up to the 12th fret, as well as the open strings with light diffusers in each hole.

The prototype contains four different parts: hardware, control app, embedded software and communications protocol. The controller is an Arduino UNO board that sends data to control the LEDs features: on/off, brightness, color, etc. To perform tests in hardware, both the Arduino and the power supply reside inside an external mdf wooden case constructed using a laser cutter. A modified XLR cable powers the PCB board and sends the data to control the LEDs. On the other hand, the control app varies in different versions of the prototype, from desktop apps, to Android and iOS apps. The control software sends the notes and chords to an Arduino UNO board containing the embedded software.

4.1 Low fidelity prototype

At first, as discussed in previous sections, the company's team participate in design methods to generate ideas for the ideal control app. Each participant tried to illustrate features and basic user interactions (Figure 3).

4.1.1 Discussion

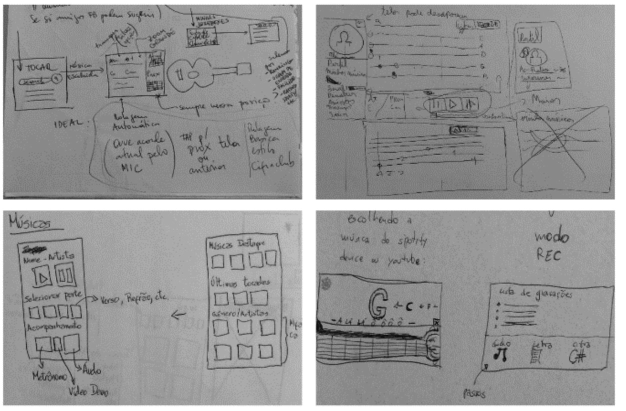

Figure 3: Low fidelity prototypes.

During this phase, the team realized the challenge in imagining the complete user interaction. The different possibilities in device orientation, number of elements displayed, highlighted elements and in deciding elements on screen and on hardware.

The main concept in this phase was the need to anticipate the chords to the user. Analysis of music learning methods and similar software like Yousician, Chordify, Guitar Hero, Guitar Pro, etc. revealed that all music notation provide a way to anticipate which notes and/or chords the user has to play. This helps the user in adjusting their posture, hand and fingers position as well as developing a mental plan of the music structures involved.

4.2 Version 0.1

In this version, the control app developed in Processing had a graphic checkbox interface to control each LED's I/O state and powered by a power supply. The system sent messages over serial protocol to the Arduino board.

4.2.1 Discussion

The possibilities in controlling each LED, its color and brightness presented way to display notes chords and music scales. However, the interaction was too cumbersome, as the user needed to select each individual note. Therefore, it became clear the need of new features that could automatically control each LED by receiving chords and notes as input.

4.3 Version 0.2

Still implemented in the same setup, this version added four modes to the instrument:

- The **song mode** is the main learning mode. A play along mode that displays the current chords on the fretboard.
- In the **scales mode** (improvisation mode), instead of chords, all possible notes included in the song's main scale were displayed. Therefore, the user

could improvise or practice solos while listening to the song even if they did not have previous knowledge of the scale’s notes.

- The **animation mode** served as performance enhancement. The user could control changes in animations by pressing different buttons or the system could automatically change animations at a certain point in the performance.
- In the “**piano roll**” mode was a demo mode that displayed the notes of selected songs without the user’s performance. The VioLED could play “Entre dos Águas” by Paco De Lucia without the musician.

4.3.1 Discussion

In this version, it became evident the main obstacles related to usability, such as anticipation of chords, occlusion and latency/jitter, as well as the need to test implementing the control software in mobile apps with wireless protocols. Furthermore, the overall memory usage of the embedded software became a concern, since it stored all the notes and chords.

Analyzing similar AMIS such as Fretlight, Fret Zealot and Populele, the main modes of operation are playing songs synchronized with the music and playing animations with the instrument. Differently from most guitar apps and AMIS such as Yousician, Guitar Pro, Fret Zealot and Fretlight, the VioLED uses a simpler chord chart instead of tablatures music notation, similar to the Populele and to music apps such as Chordify, FourChords.

Moreover, AMIS like Fret Zealot and the gtar have a distinct feature to show different colors for each finger, which shows musicians which fingers press specific notes in the fretboar. The VioLED uses a different approach showing the same color for every note, similar to the AMIS Fretlight and Populele. This considers that different colors may cause cognitive overload in the musician, since it is more information that needs to be processed every chord change. Another aspect is that different musicians may use different fingers to play the chords, choosing preferred hand positioning. However, future test is important to confirm if different colors can help or if it may cause cognitive overload.

The scales mode appears on some apps for the Fretlight, most serve as a music scale dictionary, but the Chords and Scales app also shows different scales for selected songs. However, by showing all possible notes in the scale, less experienced musicians may experience some of a “blank canvas” problem, not knowing how to begin or end musical phrases for improvisation.

4.4 Version 0.3

This version present changes in the control app. The system still utilized the core application from the previous version. However, the system used the TouchOSC iOS app as the UI (Figure 4). The TouchOSC app serves as a bridge to the Processing app, sending Open Sound Control (OSC) messages via WIFI protocol.

Figure 4: Song and Scale UI version 0.3.

4.4.1 Discussion

In this version, the control app only displayed which songs, animations and scales the user selected, without any music notation in the UI. The **song mode** implemented tempo change to help users that did not know the chord in the selected song. The user can reduce the song’s speed to 80% of the initial BPM or increase it to 120%.

This version presented a more practical UI. The interaction with mobile devices reduces the time spent to start the task as well as the ease of use by controlling the system with a multitouch screen instead of randomly assigned keyboard keys as it was in previous versions. Moreover, it revealed the need to reduce the Arduino’s memory usage, since all notes and chords for each song were in the Arduino by this version. That meant only a few songs could be implemented. Hence, the need for a new control app and a communications protocol.

4.5 Version 0.4

This version presented changes in the communications protocol, embedded app and control app. At first, the system implements a Bluetooth module (HC-06), with an Arduino microcontroller powered by a reduced mobile battery and controlled by an Android application.

Each song content was stored in JSON files over the internet and accessed by the control app. The instrument receives each chord wrapped in the communication protocol using Bluetooth (Figure 5).

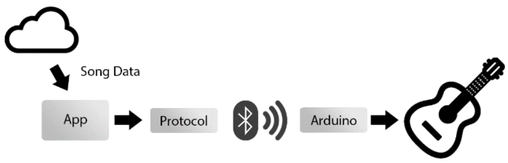

Figure 5: Version 0.4 scheme.

The communication protocol wrapper encapsulates each message sent to the VioLED in different bytes: start message, message type, message size, message (multiple bytes) and end message. Different messages can have different sizes depending on type of message (chords, metadata, color changes, etc.), chord notes, animations and others.

4.5.1 Discussion

At first, the chosen communication protocol was Bluetooth Low Energy (BLE). Despite the advantages of BLE's low latency, the 20-byte package size limitation was lacking to control animations. Since each LED uses 1 byte per color, the worst-case scenario needed 234 bytes, causing delay. Therefore, this version uses classic Bluetooth technology.

The app (Figure 6) still contains the same three modes: song mode, scales mode, animation mode. The user selects the song in a list and selects if he wants to play along (song mode) or improvise (scales mode). If the user selects song mode, the app shows the positions of the next chord as well as the current chord both in app and in the VioLED. Furthermore, the user can control the song's BPM to adjust the speed in the learning process.

Different from AMIS such as Fretlight and Fret Zealot, this version implemented chord anticipation in hardware. Before execution of each chord, an indication in red display each chord note before turning into green at the correct time (Figure 7). While most apps for the Fretlight do not anticipate chords, the Guitar Pro and the Fret Zealot relies on the anticipation of tablature notation on software. Without anticipation, the musician is surprised every chord change, since they may not know the next chord/note and its position in the fretboard. This also happens if anticipation is only in software, as the musician needs to change focus from the instrument to the app to check for changes or miss the anticipation.

The **scales mode** highlights the scale's key signature notes in red color to provide guidelines on possible beginning and end of musical phrases (Figure 8). Highlighting essential note in the scale can be a first step in attempting to solve the "blank canvas" problem.

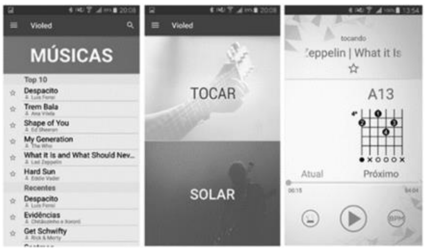

Figure 6: Version 0.4 Android app in song mode.

Figure 7: Chord anticipation in hardware.

Figure 8: Scale mode with highlighted notes.

This version still presents obstacles in usability regarding LED occlusion and the different possibilities of displaying content both in hardware and in the control app. The former is discussed in the next session. The latter relates to presenting musical information both in the instrument and in the app, which can cause cognitive overload and a bad learning experience. This suggests the need to perform tests with different versions of the system, using the app both as primary and as secondary "screen".

5. Preliminary results and discussion

The prototype's design process revealed a set of challenges and obstacles regarding different areas within the project. Most challenges involve usability aspects to display content and prototyping challenges when developing the hardware.

5.1 Obstacles in usability

Regarding content display, the identified challenges concerns time and the moment of execution of chords, occlusion of LED caused by the user's hands and feedback regarding the performance.

Preliminary results show that **chord anticipation** can help users to identify the position of next chords' notes on the fretboard. However, the system also needs to display indication of the exact moment of execution. However, fast chord/note changes may cause overwhelming sensations by constantly changing colors. Moreover, the amount of time to anticipate each chord and the means of anticipation may vary. Changing colors or blinking the notes can inform the user on the moment of execution. However, preliminary tests show that this anticipation needs to be at least on hardware, otherwise users less familiar with the song may be surprise by the notes appearing on the fretboard.

Occlusion is another obstacle in these systems. User's hand may hide additional notes or the anticipation of next notes and chords (Figure 9). This is a common problem in string instruments' AMIS. Anticipation of different notes and chords is difficult because the anticipation happens in the fretboard. However, Piano AMIS like the Yamaha Clavinova CSP [8] and the ones that use projection [10, 11, 12] solve this problem by not anticipating the notes on the actual piano keys. However, extending the visible area of the LEDs may help visibility, as in the gTar.

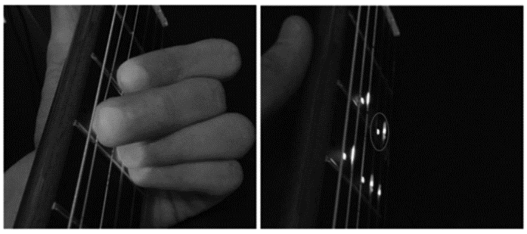

Figure 9: Occlusion cause by user's hand.

The instruments body can also occlude notes because the LED stay within the fretboard. In this case, several users may bend over the instrument to see clearly, which can compromise posture and eventually hurt users.

Possible solutions are reproducing the LEDs on the control app using computer vision, artificial intelligence, filming the fretboard and changing LEDs' position within the instrument. However, this requires further testing and hardware changes.

Another obstacle is providing reliable **feedback** by changing colors, brightness and providing audio cues from the software or from the hardware itself. As discussed in literature, complex tasks benefit from concurrent feedback, particularly in the learning phase, which can reduce cognitive overload [31]. Similar instruments and apps, i.e., Populele [7], Fret Zealot [6] and Yousician provide feedback over the user's device audio detection. The challenge is the feedback's reliability. Audio detection can compromise input and, therefore, compromise the feedback. More powerful artificial intelligence algorithms can help improve reliability at the possible cost in performance. One alternative is using sensors to detect the exact notes. However, this requires further testing since involves both hardware and software modifications.

5.2 Obstacles in hardware

When building these types of instruments, one must consider different **types of approaches**. Non-intrusive approaches may provide "sleeves" to place over the instrument, which facilitates maintenance and hardware modifications. However, the build needs to adapt to the instrument's dimensions and its limitations. As an example, when placing LEDs over the fretboard, one must consider the distances between each string, each fret and the space between the fretboard and the strings. Furthermore, placing the system over the fretboard can compromise overall ergonomics.

Contrastingly, intrusive approaches have limitations regarding hardware modifications, and system reproduction present the physical constraints of the "host" instrument. Component size and different instruments may affect the production of electronics components.

Another obstacle in hardware relates to **Latency and jitter**, which can lead to an increase in user error, affecting performance [32]. Therefore, latency also affects every choice regarding components, communication protocol and app development.

However, it is a subject fairly studied in the literature. Some authors provide guidelines on different usages and types of control [20, 28], discussing that highly demand systems may require latency below 10ms. When displaying notes and chords at execution time, the AMIS may require low latency or it could hinder synchronization, chord anticipation and, eventually, the overall performance.

Energy consumption also presents an obstacle in the development process. The selected RGB LEDs (WS2812B) operate in 5V and require 20mA per channel (60mA per LED on RGB). Therefore, the system requires approximately 5A to display every one of the 78 LEDs on white color.

In current version of the VioLED, the system applies a normalization of the color bytes. Generally, the LED work with one byte per channel (0-255).

$$i_{system} = \sum_{n^{o}\ LEDs} i_{LED}$$

Equation 1: Equations for overall system current and for each LED.

The normalization limits the sum of channel components to 120 instead of 765. The normalization reduces the overall consumption to approximately 16% of the worst-case scenario, leading to power supply reduction and system portability.

6. Conclusions and future research

This paper presented a study of Augmented Musical Instruments for Study (AMIS). The study shows that the design of these AMIS involves challenges regarding specific computer science areas as well as challenges music learning.

Using research through design method, a prototype is currently on development and the process behind developing its early versions already provided insights on usability and prototyping aspects. These relate to anticipation of chords, user feedback, latency and communication protocol requirements, minimum energy consumption, occlusion, etc. as well as generating hypotheses on possible solutions, which could guide future tests in similar instruments.

Chord anticipation and highlighting essential notes in music scales are important features in the learning process that current AMIS do not implement in hardware. On the other hand, occlusion still is presents a, obstacle in the overall experience. Furthermore, the developed prototype still needs further improvement in feedback, as well as communications protocol and energy consumption optimization compared to AMIS in the market.

The development process corroborates with the challenges found in the literature. Each change in hardware/software affects the system as a whole and require further testing regarding latency, energy consumption, usability and tests with subjects to provide benefits to the learning experience.

Concerning future works, current versions of the control app, in development, implement features such as current chord duration, different formats to display content, and starts to integrate social network elements to share user's performance. The main concern is in how to display content between instrument and app in order to provide a better experience to the musician.

Results in preliminary user tests with 22 subjects in the Semana Nacional de Ciência e Tecnologia (SNCT) event suggest that the user's experience improved when learning new songs with the VioLED system when compared to traditional chord charts and the control app with a traditional acoustic guitar. The subjects noted that they felt "more confident" when using the VioLED. However, it is important to perform further testing with different groups and different setups to improve the system and produce design guidelines for similar systems.

References

[1] Aumeo. (n.d.). https://aumeoaudio.com/.

[2] E. R. Miranda and M. M. Wanderley. (2006). *New digital musical instruments: control and interaction beyond the keyboard* (Vol. 21). AR Editions, Inc.

[3] L. Turchet, C. Fischione and M. Barthet. (2017). Towards the Internet of Musical Things.

[4] K. L. Lueth. (2020). *State of the IoT 2020: 12 billion IoT connections, surpassing non-IoT for the first time*. IoT Analytics - Market Insights for the Internet of Things. https://bit.ly/3yrD2ac.

[5] The Fretlight Guitar Store. (n.d.). *The Fretlight Wireless Guitar Store*. Retrieved August 6, 2021, from https://fretlight.com/

[6] Fret Zealot. (2021). *Best Way To Learn Guitar | How To Learn Guitar At Home*. https://www.fretzealot.com/

[7] Populele - Smart Ukulele (n.d.). *Populele Smart Ukulele by PopuBand Music Inc*. Retrieved August 6, 2021, from https://popuband.com/

[8] Yamaha Clavinova CSP. (n.d.). *Clavinova CSP*. Retrieved August 6, 2021, from https://yamaha.io/3yujPED

[9] S. Seol, Y. Shin and K. Lee. (2016). Learning Guitar with an Embedded System. *Contemporary Engineering Sciences*, *9*(12), 553-560.

[10] M. Löchtefeld, S. Gehring, R. Jung, and A. Krüger, guitar: Supporting Guitar Learning through Mobile Projection, CHI 2011 Extended Abstracts on Human Factors in Computing Systems, (2011), 1447-1452.

[11] L. Raymaekers, J. Vermeulen, K. Luyten and K. Coninx (2014). *Game of tones: Learning to play songs on a piano using projected instructions and games*. In CHI'14 Extended Abstracts on Human Factors in Computing Systems (pp. 411-414).

[12] Music Everywhere. *Music Everywhere - Instant Musician AR Piano for iPhone*. (n.d.). Retrieved August 6, 2021, from https://www.music-everywhere.co/home

[13] T. Issa and P. Isaias (2015). Usability and human computer interaction (HCI). In *Sustainable design* (pp. 19-36). Springer, London.

[14] Dix, A., Finlay, J. E., Abowd, G. D., & Beale, R. (2003). *Human-Computer Interaction* (3rd ed.). Pearson.

[15] Nielsen, J. (2012). *Usability 101: Introduction to Usability*. Nielsen Norman Group. https://bit.ly/3ysRyhQ.

[16] K. Y. Huang (2009, October). *Challenges in human-computer interaction design for mobile devices*. In Proceedings of the World Congress on Engineering and Computer Science (Vol. 1, No. 6, pp. 1-6).

[17] G. De Michell and R. K. Gupta (1997). Hardware/software co-design. *Proceedings of the IEEE*, *85*(3), 349-365.

[18] D. Chen, J. Cong, S. Gurumani, W. M. Hwu, K. Rupnow and Z. Zhang (2016). *Platform choices and design demands for IoT platforms: cost, power, and performance tradeoffs.* IET Cyber-Physical Systems: Theory & Applications, *1*(1), 70-77.

[19] M. M. Wanderley and N. Orio (2002). *Evaluation of input devices for musical expression: Borrowing tools from hci*. Computer Music Journal, *26*(3), 62-76.

[20] S. Jordà (2005). *Digital Lutherie Crafting musical computers for new musics' performance and improvisation*. Department of Information and Communication Technologies.

[21] C. Palmer and R. K. Meyer (2000). *Conceptual and motor learning in music performance*. Psychological Science, 11(1), 63-68.

[22] N. MIKROS (2004) *Using Software Prototypes In Game Design*. In Game Design Workshop–Designing, Prototyping, and Playtesting Games. [s.l.] CMP Books, 2004. p. 460.

[23] S. Oore (2005). *Learning advanced skills on new instruments*. In Proceedings of the 2005 conference on New interfaces for musical expression (pp. 60-64).

[24] G. McPherson (Ed.). (2015). *The child as musician: A handbook of musical development*. Oxford University Press.

[25] H. Pinto (1978). *Iniciação ao violão*. São Paulo: Ricordi, 1.

[26] A. C. N. do Couto (2009). Música popular e aprendizagem: algumas considerações. *Opus*, *15*(2), 89-104.

[27] S. Jordà (2004, June). *Digital instruments and players: part I---efficiency and apprenticeship*. In Proceedings of the 2004 conference on New interfaces for musical expression (pp. 59-63).

[28] D. Wessel and M. Wright (2002). *Problems and prospects for intimate musical control of computers*. Computer music journal, 26(3), 11-22.

[29] J. Zimmerman, J. Forlizzi and S. Evenson (2007). *Research through design as a method for interaction design research in HCI*. In Proceedings of the SIGCHI conference on Human factors in computing systems (pp. 493-502).

[30] D. Fallman (2003). *Design-oriented human-computer interaction*. In Proceedings of the SIGCHI conference on Human factors in computing systems (pp. 225-232).

[31] R. Sigrist, G. Rauter, R. Riener and P. Wolf (2013). *Augmented visual, auditory, haptic, and multimodal feedback in motor learning: a review*. Psychonomic bulletin & review, 20(1), 21-53.

[32] I. S. MacKenzie and C. Ware (1993). *Lag as a determinant of human performance in interactive systems*. In Proceedings of the INTERACT'93 and CHI'93 conference on Human factors in computing systems (pp. 488-493).