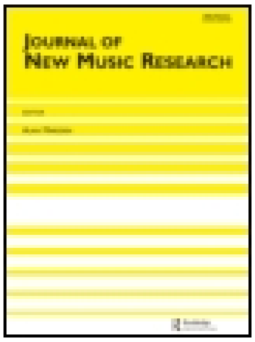

Journal of New Music Research

ISSN: 0929-8215 (Print) 1744-5027 (Online) Journal homepage: http://www.tandfonline.com/loi/nnmr20

PLXTRM: Prediction-Led eXtended-guitar Tool for Real-time Music applications and live performance

Tim Vets, Jonas Degrave, Luc Nijs, Federica Bressan & Marc Leman

Journal of New Music Research, 2017
http://dx.doi.org/10.1080/09298215.2017.1288747

PLXTRM: Prediction-Led eXtended-guitar Tool for Real-time Music applications and live performance

Tim Vets[1], Jonas Degrave[1], Luc Nijs[1], Federica Bressan[2] and Marc Leman[1]

[1]*Ghent University, Belgium;* [2]*University of Padova, Italy*

(Received 22 June 2016; accepted 11 January 2017)

Abstract

This article presents PLXTRM, a system tracking picking-hand micro-gestures for real-time music applications and live performance. PLXTRM taps into the existing gesture vocabulary of the guitar player. On the first level, PLXTRM provides a continuous controller that doesn't require the musician to learn and integrate extrinsic gestures, avoiding additional cognitive load. Beyond the possible musical applications using this continuous control, the second aim is to harness PLXTRM's predictive power. Using a reservoir network, string onsets are predicted within a certain time frame, based on the spatial trajectory of the guitar pick. In this time frame, manipulations to the audio signal can be introduced, prior to the string actually sounding, 'prefacing' note onsets. Thirdly, PLXTRM facilitates the distinction of playing features such as up-strokes vs. down-strokes, string selections and the continuous velocity of gestures, and thereby explores new expressive possibilities.

Keywords: augmented guitar, music performance, jazz improvisation, gesture analysis, music education

1. Introduction

The advent of computers capable of processing audio with low latency has stimulated research on augmented music instruments; these are traditional instruments onto which additional technology (hardware and software) is added so that the instrumental playing, and/or its sonic output reaches sound processing beyond what is normally possible on the traditional instrument. Miranda and Wanderley (2006) state that 'the original instrument maintains all its default features in the sense that it continues to make the same sounds it would normally make, but with the addition of extra features that may tremendously increase its functionality' (p. 22). Refer to Miranda and Wanderley (2006) also for examples of augmented piano, strings, saxophone, trumpet, flute and so on. We will call these additional extra features that increase the traditional instrument's functionality: the 'augmentation'.

In this context, musicians are confronted with the question of how augmentation affects playability and expressiveness. Often, musicians feel that augmentation requires a new playing technique in order to control the expressive possibilities offered by added electronics. Augmentation may indeed affect the control of expressiveness at the micro-timing level. By adding electronics, control may appear less direct, less based on haptic feedback and physical causation. There is also a risk of adding excessive cognitive load, due to augmentation, which distracts the musician's attention from expressive control. In other words, augmentation is possible, and feasible with low latency technology, but a major question concerns the integration of the augmentation with human interactive capabilities. What we claim here is that human sensorimotor control of musical expression may gain a lot when augmented features are made anticipative with respect to the human control capabilities.

Based on recent insights in embodied music cognition (Leman, 2007; Maes, Leman, Palmer, & Wanderley, 2014; Maes, Nijs, & Leman, 2015) and expressive music interaction (Leman, 2016) we believe that the musical end result of research on augmented music instruments can benefit from a better integration of the instrument's electronic sound processing with the musician's sensorimotor control during the performance. Improving the control of expressive interactions has the potential to unlock the sonic possibilities of the electronic sound processing in relation to human action and its associated predictive processing.

In the present paper, we propose a system for extended guitar that allows the prediction of string picking. The information

Correspondence: Federica Bressan, Department of Information Engineering, University of Padova Via G. Gradenigo 6/B, 35131 Padova, Italy. Email: federica.bressan@dei.unipd.it

obtained by 'looking' a few milliseconds ahead of the actual playing is then used for predictive sound processing ('preambles') and more accurate expressive control (e.g. based on probabilities of an onset). The structure of the paper is as follows. In Sections 2 and 3 respectively, we present background information about augmented guitars and the concept behind PLXTRM. In Sections 4 and 5 we explain how PLXTRM has been implemented (hardware and software) and how its predictive capacities have been evaluated. In Section 6 we explain the expressive effects obtained by prediction, and in Section 7 we discuss them in the light of the concepts from research in embodied music cognition.

2. Augmented guitars

Electronically and digitally augmented guitars have been around since the early 1980's (e.g. Polson, 1982). A notable example is the use of pitch tracking allowing the guitar to control Musical Instrument Digital Interface (MIDI) capable sound sources. Commercial MIDI-guitar products are aimed at tracking the played pitches and return a symbolic representation of played notes in the form of MIDI note, velocity and possibly pitch-bend or modulation values. With optimised tracking algorithms, some of these applications obtain good results even when applied to the monaural audio signal of the guitar that contains a combination of all six strings (e.g. von dem Knesebeck & Zölzer, 2010). Some MIDI guitar systems have employed ultrasound technologies to achieve this goal (e.g. Fala, Keshap, Doerning, & Barbeau, 1991). Lastly, a *hexaphonic pickup* is often used for MIDI guitar applications, in order to split the polyphonic signal to be tracked into six monophonic signals. The aforementioned applications track played notes, but do not track the activity of the guitar pick itself, and they are not predictive. The instrument augmentation with PLXTRM bears resemblance to other augmented instruments in that it extends the instrument's sonic possibilities while retaining its playability with the traditional techniques (Lähdeoja, 2015).

The potential of augmented guitar picks has been explored in the past by some academic as well as commercial projects. In the MIDI pick, Vanegas (2007) places a pressure sensor on the guitar pick, providing the player with a continuous controller. The controller has to be used by varying the pressure of the fingers on the guitar pick. This requires the player to use gestures that are not part of conventional guitar playing, which may introduce undesirable distraction from the performance. The Air Guitar Move for iPhone[1] is an entertainment product consisting of a guitar pick that can be connected to an iPhone and that enables the user to play air guitar. Unlike in the system presented in this article, the Air Guitar Move is not intended to be used with a real guitar. The Firefly pick (Holm & Williams, 2014), a Kickstarter campaign from 2014, is used to produce a light show by flashing LED's within the guitar pick in response to strumming motion. Yet the processing does not impact the produced sound: conversely, the system presented in this article aims to extend the expressive range of the guitar for musical purposes. Finally, the gXtar project by Kessous, Castet, and Arfib (2006) contains an augmented guitar pick, composed of a piezo film sensor fixed between two thin plectra with glue. Time-frequency analysis of the piezo film signal was then used to extract intensity of playing. This piezo-based approach uses pressure sensing only, and therefore does not take into account the trajectories of the guitar pick during in-air motion. Conversely, PLXTRM gives a more complete basis for extracting information from the motions of the guitar player by combining touch detection and motion detection. Lastly, some projects, such as Larsen, Overholt, and Moeslund (2013), attempt to automate the physical actions of the guitar pick altogether by motorizing it. With respect to the aforementioned examples, PLXTRM's distinguishing feature is clearly represented by *prediction*, as additional tool for musical expressiveness. The next Section describes the concept behind the system.

3. The PLXTRM concept

The main source of the timing in guitar playing lies in the picking-hand. To obtain sound manipulations that preserve the integrity of musical timing, our strategy is to use information retrieved from as close as possible to that timing-source: the guitar pick (the cause), rather than the sounding notes (the effect). Our goal is to prepare for the guitar pick in such a way that augmentation can use this anticipatory information in view of expressive sound processing.

3.1 Sound processing and expression

There seems to be a trade-off between latency, control and expanded sonic possibilities. In order to create a highly personal musical language, electronic musicians expect flexibility from the tools they use. Therefore, patching environments such as Pure Data, Max MSP or Reaktor are often used for sonic augmentation. These environments run on general purpose operating systems, and standard computing hardware (e.g. a laptop running OS X, Windows or Linux, external sound card for improved audio fidelity, increased input / output channels, and reduced latency). Commonly available computer and sound card setups are capable of running such patching environments at low latencies for live performance use. Latencies of 5–10 ms are commonly achievable, depending on the complexity of the audio signal processing, and the background load (McPherson, Jack, & Moro, 2016). However, even at these low latencies, expressive control is still less direct than playing an non-augmented traditional guitar. While reducing latency in live electronic instruments is an important topic of research in itself, we approach the issue from a different perspective, by investigating what can be done with events that are predicted before they occur.

[1] http://airguitarmove.com.

3.2 Expression and prediction

Human expressive interaction is strongly linked with prediction processes (Leman, 2016). Of particular relevance are sensorimotor brain processes that control the relationship between actions and sensory outcomes of those actions (Maes et al., 2014). Plucking the string is a typical example of an action, and the proprioceptive (or kinesthetic) feedback, sensed tactile, sonic, and perhaps visual feedback are typical consequences of that action. Sensorimotor processes thus regulate the association of sensory outcomes of actions with the action itself. These processes are fast and operate at the millisecond level. The expressive control based on these predictive processes is accurate and an intervening technology (such as based on augmented electronics, in particular sound processing) should not alter the sensorimotor predictions for expression, because that would affect the control of these predictions. Further motivation for our approach is found in the theory of enactment, in particular its role in prediction aspects of the music playing process. This theory is also employed by the 'new tangible musical instruments' described in Essl (2006), which center on the 'maintenance of familiar sensorimotor experiences in interactive settings corresponding to sounding phenomena'. If sound processing could already be prepared before the string is going to be plucked, that would allow augmented sound output in time with the human sense of expressive micro-timing. An example of such a predictive technology, although in another domain, is D-Jogger (Moens et al., 2010; Moens et al., 2014; Moens and Leman, 2015). This technology predicts the time at which the foot will touch the ground during walking or running, based on previous occurrences of footfalls. This information is needed in order to be able to provide the music in such a way that the time of the musical beat exactly matches the time at which the foot touches the ground. In many respects we want this guitar system to act in a similar way: it should predict the human plucking action in order to provide sound processing in time, or even ahead, with this action. However, the augmented guitar differs from the D-Jogger application in that we cannot rely on a periodic movement. But we can rely on the sensorimotor preparation for action, which in this case can be measured in the movement of the guitar pick, for example, when it moves towards one of the strings. In the setup described in this article, the main audio processing is performed by Pure Data, running on a system with a 5 ms latency. However the latency can be reduced to 2.04 ms running the audio processing on the Axoloti board (see Section 4.5). This means that, when a note onset is predicted 20 ms beforehand, the audio latency must be subtracted, making it in practice a 17.96 ms prediction. To sum up, our goal is to fully integrate the augmented sound processing technology with the sensorimotor predictive processing capabilities of the player, so that the expressive control feels more direct.

4. System implementation

The PLXTRM system consists of 5 interconnected elements.

- guitar pick with electrical contacts, accelerometer and piezo sensor
- hexaphonic pick-up and pre-amp
- string identification
- signal conditioning front end for the ADC integrated in the micro-controller
- Reservoir and DSP on Axoloti Core board

4.1 Guitar pick

The guitar pick used to play the electric guitar is commonly a non-sensorized object made of materials such as plastic, rubber nylon or Polyamide-imide.[2] Although the pick can be used in many different ways, we will focus here on a usage common in jazz-guitar soloing. The pick is held in the strumming hand between the thumb and index finger, and used to select and play consecutive notes with expressive timing and dynamics. The PLXTRM pick is based on a pick of 3 mm thickness ('extra heavy', Big Stubby, Jim Dunlop[3]). The concept arose from the observation that, in conventional guitar picking, only a limited set of gestures features performed by the musician contribute to the final production of the sound. There is another set of gesture-characteristics that can be exploited for additional sound control, the potential of which is explored by the application presented in this paper. These gestures have also previously been referred to as 'ancillary gestures', as opposed to 'sound-producing gestures', 'sound-accompanying gestures' and 'communicative gestures' (Lähdeoja, Wanderley, & Malloch, 2009, p. 327). Other applications have attempted to extend the guitar's expressive possibilities by adding extrinsic gestures (i.e. gestures that are unrelated to the sound production, and are neither functional to 'accompanying' or 'communication' with other players or with the audience). One key feature of the authors' approach is that gestures intrinsic to guitar-playing are used in new ways for creating sound. Several aspects of the gesture vocabulary of jazz-guitar playing, of which the impact on the resulting music is less obvious, can be used to control the production of sound. The main question that PLXTRM investigates in this respect is how they relate to musical expression, and how to use them in new ways to create new expressive possibilities in music performance.

A closer look at the picking action reveals 4 distinct phases (see Figure 1).

- phase 0: pick moves towards string
- phase 1: pick-to-string impact
- phase 2: pick-to-string release, string is pushed aside and released from pick

[2] http://www.dragonsheartguitarpicks.com/design/.

[3] http://www.jimdunlop.com/product/stubby.

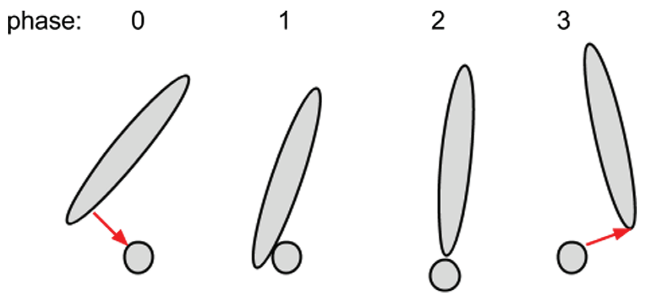

Fig. 1. The four phases of guitar pick stroke. Before the pick touches the string (0), contact with the string (1), right after plucking the string (2), the pick moves away from the string (3).

- phase 3: string vibration onset, pick moves away from string

In phase 0, where the pick moves towards the string, no sound is produced, yet the musician has decided on the musical action of playing a particular note. In phase 1, the pick collides with the string. This is the first point where a clearly timed event is detectable. Phase 2 is where the pick has been pushed against the string so that the string has moved sideways, and eventually escapes from the pressure applied by the pick. This phase is also called the 'final slip-off' in Perng, Smith, and Rossing (2011). This is the crucial moment where the string starts to vibrate. The release of the string causes a second peak in the pick which in some cases can be detected as a separate onset with a piezo sensor. With the electrical contacts on the guitar pick, this is the moment where the voltage goes back to zero. In phase 3 the pick moves away from the string and the vibration of the string has started. This is where onset detection on the audio signal takes place.

Traditionally, guitarists use foot-switches to change parameters in the effects chain. Also, several continuous controllers are traditionally controlled with foot pedals (e.g. volume control, wah-wah, whammy pedal…). When playing while standing, there is only one foot available for any of these types of control devices. The added possibility of using hand gestures without requiring moving the hand away from the strings opens new possibilities to change aspects of the sound while playing, and using a foot controller simultaneously.

The electrical contacts (see Figure 2(A)) are copper plates attached to the front and back of the pick. They are normally-open (N.O.) contacts used to measure voltage when touching the guitar strings (i.e. closing a contact) (see Section 4.3). A separate contact on the front- and backside of the pick allows distinguishing up-strokes from down-strokes.

The accelerometer is mounted in the centre, near the wide side of the pick, in the area covered with rubber shielding (see Figure 3). The shielding is needed to prevent the hand holding the pick from creating false contact on the copper contacts as well as on the accelerometer circuitboard. Lastly, a small piezoelectric disk is attached to the guitar pick, as a second way of sensing the pick-to-string collisions. This sensor gives a continuous signal, as opposed to the electrical contacts, which yield discrete a signal (on/off). As such, the piezo sensor provides more information about the impact velocity (dynamics) of played notes. It is also used for generating training data for the reservoir. Training the reservoir with the electrical contacts alone, leaving out the piezo sensor, decreased prediction accuracy dramatically. The piezo signal was converted to a DC signal with the signal conditioning circuit described in Section 4.4 to be connected to a GPIO port on the Axoloti board.[4]

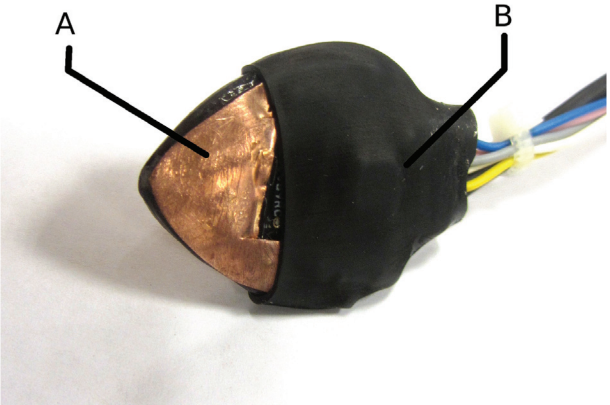

Fig. 2. PLXTRM guitar pick with copper contacts (A) and embedded accelerometer (B).

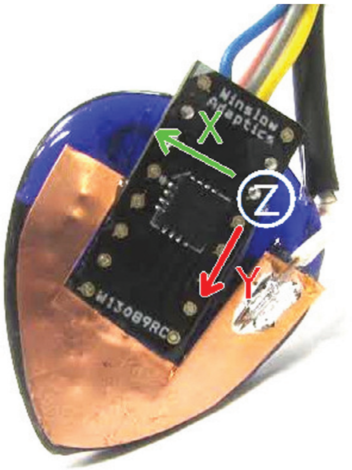

Fig. 3. The PLXTRM accelerometer. The x-axis corresponds to the left/right movement of the hand holding the guitar pick, the y-axis to the movement from and to the player's body (distance from strings, on horizontal plane), the z-axis to the up/down movement.

[4] Axoloti is a platform that blends sketching of digital audio algorithms with the musical playability of standalone hardware. It is an open source project by an independent Belgian developer. See Section 4.5.

4.2 Hexaphonic pick-up and pre-amp

A custom-built hexaphonic guitar pickup was used, similar to the Roland GK3 pickup, which provides a separate audio signal for each guitar string. The custom pickup has larger coils and magnets, to achieve a personal sound that differs from what commercial hexaphonic pickups provide. The signal of the hexaphonic pickup is pre-amplified by a hexaphonic pre-amp based on the OPA2134 High Performance Audio Operational Amplifier.[5] The high-impedance guitar audio signals are thus converted into a line-level audio signal that can be used in common audio processing setups. This hexaphonic pickup and preamp were previously created for the extended-guitar improvisations project.[6]

4.3 String identification

To achieve this, the guitar was modified to isolate each string electrically from all other strings, turning each string into a separate conductor (see Figure 4). Five modifications have been made to the guitar bridge and tailpiece.

(1) The anchoring bar where the strings are normally anchored to the tailpiece is replaced with a wooden bar (Figure 4(A) and (B)).
(2) The wooden anchoring bar to which the strings are anchored is attached to the original anchoring bar by two coated steel cables. The coating on these cables provides electrical isolation (Figure 4(B)).
(3) The intermediate bar, which pushes down the strings to create the tension keeping them on the saddles (Figure 4(D)) is isolated using a piece of polyvinyl chloride tubing (PVC pipe) (Figure 4(C)).
(4) Three types of contact cable clamps were used. Cylindrical screw clamps were found to be more suitable for a permanent solution, because they are guaranteed to stay on the strings. Removing these clamps does however require the strings to be removed from the guitar. Therefore clamps of the third type (strings one and two, on the rightmost side of Figure 4) are more suitable for temporary installation of the system.
(5) The bridge saddles are isolated by inserting a thin plastic layer between the strings and the saddles (Figure 4(E)). It is thin enough to have negligible influence on the vibration of the strings, and it is easily replaceable.

From the first string to the sixth, the voltages are distributed via a resistor ladder that receives 3.3v from the Axoloti board. The strings carry the following voltages: string 1 (high E): +2.45 V, string 2 (B): +2.11 V, string 3 (G): +1.47 V, string 4 (D): +0.95 V, string 5 (A): +0.8 V, string 6 (E): +0.32 V. By measuring the voltage when the pick touches a string, it can be determined which of the 6 strings was chosen.

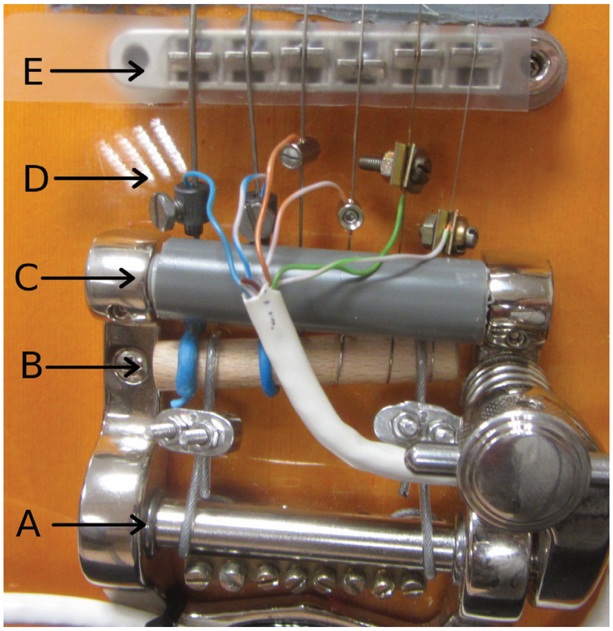

Fig. 4. The PLXTRM modified bridge with electrically isolated strings. Letter A indicates the strings anchoring bar, which was replaced by a wooden bar (letter B). Letter C indicates the intermediate bar, which is electrically isolated. Letter D indicates the three types of clamps. Letter E indicates the thin plastic layer between the strings and the saddles, providing electric isolation.

During an earlier stage of development, PLXTRM achieved string identification in a different way. Instead of placing voltages on the strings, two distinct voltages were placed on the guitar pick (3.3 V on the down-stroke side, and 1.65 B on the up-stroke side). The six strings were then connected to a separate General Purpose Input/Output (GPIO) port each, and measured the incoming voltages from the pick. This brings the total number of GPIO channels to be processed for this method to 8. Because frets on a regular electric guitar are conductive, when several strings are pressed with the left hand, all pressed strings will be electrically connected. As a result, all the pressed strings are reflected at once in the tracking result, as soon as any one of them is touched with the pick. This means that when the first string in a chord is picked, the system can simultaneously identify the other strings that make up the chord. In some cases of polyphonic playing, this method allowed predicting pick-to-string collisions several steps beforehand. In the final version we chose the more efficient solution of placing distinct voltages on the strings and detect those with the contacts on the pick, although this configuration may still prove useful for applications related to polyphonic playing. The current solution brings the advantage of focussing on single-note solo playing styles, as well as requiring only 2 GPIO channels instead of 8. Edge cases where the guitar pick touches two strings at once are rare, especially given the single-note playing styles PLXTRM is focussed on.

[5] http://www.ti.com/lit/ds/symlink/opa2134.pdf.
[6] www.timvets.net/music/egi.

Moreover, for any user using PLXTRM, a new profile can be made by recording new training data for generating weights for the reservoir. Therefore, should a user have a particular use case where touching several strings at once with the pick is necessary and happens frequently, the prediction system will adapt to that. It should be noted however that touching two strings simultaneously might result in the upstroke and downstroke sensors on the pick to be activated at the same time, making the up/down stroke distinction unreliable. At the voltage detection update speed of 3kHz, transitions between strings do not cause issues with ambiguous voltage states.

4.4 Signal conditioning front end for the ADC integrated in the micro-controller

In addition to the pick-to-string collision detection via the copper contacts on the PLXTRM pick, regular audio signal based onset detection is done on the audio signals of the strings. Onset detection on the sounding notes of the strings was used to measure the time difference between the onsets predicted by the reservoir, pick-to string collisions and sound-based onset detection, as well as for generating training data for the reservoir. To do this, we need 6 audio channels (one for each guitar string). Since the Axoloti has only two audio in- and outputs, we created the signal conditioning boards in order to connect the audio signals to 6 Axoloti GPIO ports. In terms of audio quality this does not provide a signal that can be used directly for audio processing, since the sampling rate of these GPIO pins is 3kHz, but for the purpose of string-by-string onset detection it is sufficient. Moreover, it allows maintaining a small physical form-factor of the system, eliminating the need for 6 Analog To Digital Conversion channels. The audio signals of the six strings need to be prepared for being read as DC signals. An signal conditioning circuit board 5 was designed, based on the LM324 quadruple operational amplifier.[7]

Each board boasts 4 channels, and takes in a regular line level audio signal from the hexaphonic pre-amp. The ranges of the audio signals are then converted from a zero biased signal (-1.095–$+1.095$ V) to a signal ranging between 0.0v and 3.3v, by scaling it and adding a bias of $+1.65$ V.

4.5 Axoloti core

Axoloti is a platform based on the 168 MHz STM32F4 micro-controller for the sketching of digital audio algorithms. Figure 6 shows the front view of an Axoloti core v1.0. It was chosen for our application as a replacement for a laptop/soundcard setup, because of the small form factor, the low latency performance, and its compliance with current trends in the field of preservation of computer based systems for music performance (e.g. Axoloti is an open system: hardware and software) Bressan (2014). With its audio sample rate of 48kHz and a buffer size of 16 samples this latency amounts to 0.333 ms. Taking into account the analog-to-digital and digital-to-analog conversion latency, its measured latency is around 2.04 ms.[8] It also has all the infrastructure required for PLXTRM on board; GPIO pins, Audio input/output, MIDI input/output all integrated into one development environment. Once a program is created for it, the Axoloti Core board can run it as standalone hardware. Developing audio applications on the Axoloti platform is done in a patching environment similar to Max/MSP,[9] Pure Data[10] or Clavia Nord Modular G2's Patch Editor.[11]

4.6 Reservoir computing

In order to combine the input sensors into making a prediction of a stroke before it occurs, we used the Reservoir Computing paradigm (Schrauwen, Verstraeten, & Van Campenhout, 2007; Lukoševičius & Jaeger, 2009). In this paradigm, the input signals are first combined and mixed with each other in a non-linear way, in a near-chaotic system. This increases the richness of the input to a higher number of signals, in which it is easier to find a system which correlates to the desired output. This can be done outside of processors, using for instance photonic devices (Larger et al., 2012; Paquot et al., 2012) or even a bucket of water (Fernando & Sojakka, 2003). As long as the dynamical system used to mix up the input signals has the mathematical properties described in Lukoševičius and Jaeger (2009), the system is fit for pre-processing the input signals. After this pre-processing of the input signals, a simple linear method can be used to discover even non-linear relationships between the input data.

In the PLXTRM application, we used the more common recursive neural network (RNN) approach to reservoir computing. From the PLXTRM, we receive four input signals at a 3kHz sample rate to generate a prediction of an onset, namely the acceleration on 3 axis and a signal from the piezo element. These input signals $\boldsymbol{x}^t$ at time t are then linearly projected with randomly chosen matrices $\boldsymbol{W}_{\text{hidden}}$, $\boldsymbol{W}_{\text{input}}$ and $\boldsymbol{w}_{\text{bias}}$ before being non-linearly transformed using a tanh activation function.

$$\boldsymbol{x}^t_{\text{pre}} = \tanh(\boldsymbol{W}_{\text{hidden}} \cdot \boldsymbol{x}^{t-1}_{\text{pre}} + \boldsymbol{W}_{\text{input}} \cdot \boldsymbol{x}^t + \boldsymbol{w}_{\text{bias}})$$

Consequently, these preprocessed signals are then classified using linear regression to approximate the desired output signal $\hat{\boldsymbol{y}}^t$, namely an estimation of the chance a stroke will happen in the next 50 ms.

$$\hat{\boldsymbol{y}}^t = \boldsymbol{W} \cdot \boldsymbol{x}^t_{\text{pre}} + \boldsymbol{w}$$

In this equation, we need to optimize the parameters $\boldsymbol{W}$ and $\boldsymbol{w}$ in order for $\hat{\boldsymbol{y}}^t$ to have the desired behavior. Note that this estimation $\hat{\boldsymbol{y}}^t$ can and does take values outside of the interval [0, 1].

[7] http://www.ti.com/product/lm324.

[8] http://www.axoloti.com/more-info/latency/.

[9] https://cycling74.com/.

[10] http://puredata.info/.

[11] http://www.nordkeyboards.com/.

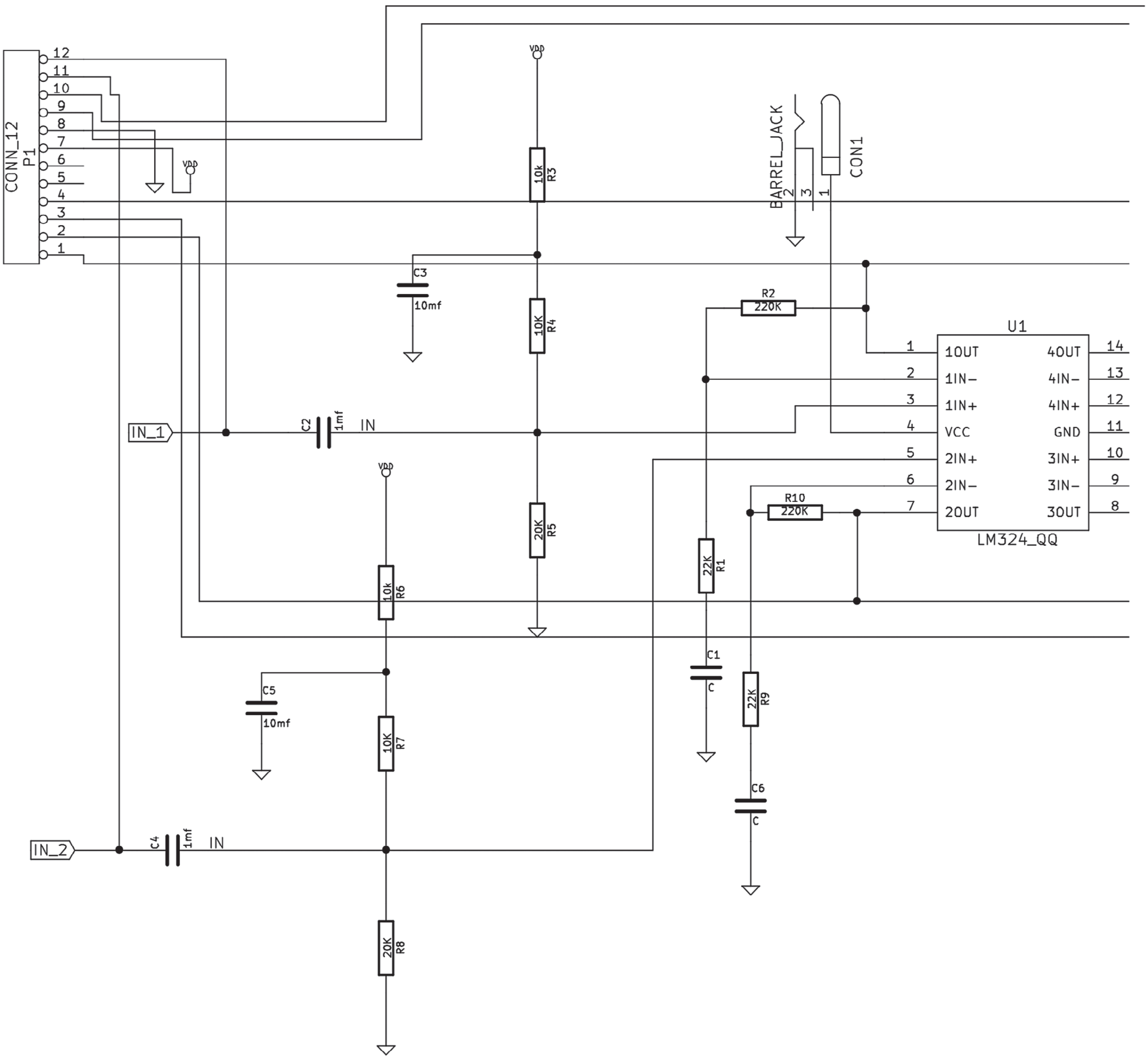

Fig. 5. Schematic for 2 channels of the signal conditioning board used for PLXTRM.

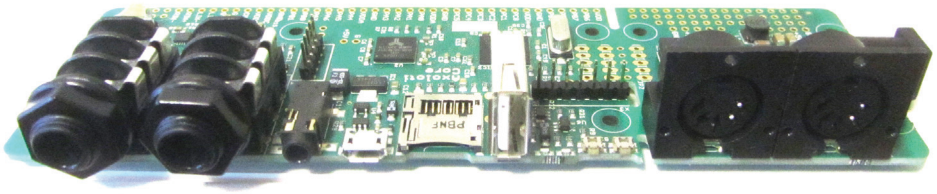

Fig. 6. Front view of the Axoloti Core v1.0, featuring: stereo audio DAC/ADC, headphones plug, USB device port, USB host port, MicroSD socket, MIDI in and MIDI out. PLXTRM exploits the GPIO ports on the back side of the board to connect the guitar string and the pick accelerometer.

In our case, we needed to minimise the necessary calculations in both these formulas during evaluation time. To do this, we had to make various adjustments to the common implementations of these algorithms.

- We used a piecewise linear function to approximate the tanh activation function. As long as this piecewise linear function is output bounded and non-linear, it suffices for reservoir computing. The compiler used to program an Axoloti can expand this piecewise linear function into efficient machine code. This makes the binary code faster than a native tanh implementation.
- We used sparse matrices $\boldsymbol{W}_{\text{hidden}}$, $\boldsymbol{W}_{\text{input}}$ and $\boldsymbol{w}_{\text{bias}}$ to minimise the number of multiplications needed to preprocess the input signals. As long as the mathematical properties of the dynamical system remain unchanged, this does not affect the performance of the reservoir.

In order for this system to work, we need to find the matrices $\boldsymbol{W}$ and $\boldsymbol{w}$ which will do a linear transformation between the preprocessed input signals and the desired output signal. This is now easy to obtain using linear regression.

We recorded the input signals from the accelerometer and the piezo element, together with the onset detection. Using this onset detection, we created a ground truth matrix $\boldsymbol{y}$ of the desired output. This matrix has the value 1 when an onset will be detected within 50 ms, and a 0 otherwise. The input signals were normalised such that they have a mean of zero and a standard deviation of one, and a bias signal was added to obtain the input matrix $\boldsymbol{x}$. The bias signal is necessary in order to reduce the search for the matrices $\boldsymbol{W}$ and $\boldsymbol{w}$ to a single matrix $\boldsymbol{W}$.

Using these matrices, the matrix which minimises the mean squared error (MSE) between $\hat{\boldsymbol{y}}$ and $\boldsymbol{y}$ on the training data can simply be found using the linear regression normal equation:

$$\boldsymbol{W} = (\boldsymbol{x}^T \boldsymbol{x} + \lambda \boldsymbol{I})^{-1} \cdot \boldsymbol{x}^T \cdot \boldsymbol{y}$$

In this equation, there is a regularisation parameter λ which determines the amount of generalisation done on the input data. We found that a value of 2×10^{-6} for λ minimised the MSE on a validation set.

Finally, in order to run this RNN on the Axoloti, we had to optimise the number of nodes in the hidden layer N, in our case the length of the matrix $\boldsymbol{x}^t_{\text{pre}}$, and the sparsity of the matrices $\boldsymbol{W}_{\text{hidden}}$, $\boldsymbol{W}_{\text{input}}$ and $\boldsymbol{w}_{\text{bias}}$. We found that the RNN using $N = 50$ and a sparsity of 0.04 minimised the MSE on the validation set, while still being able to run on the Axoloti.

5. Evaluation of PLXTRM predictions

This reservoir can now be used to predict when an onset will happen in the next 50 ms, namely when $\hat{\boldsymbol{y}}^t > 0.5$. In order to establish the quality of the predictions, we looked at the distribution of the predictions vs. our ground truth, which is the real onset detection on the signals of the hexaphonic preamp.

To evaluate our setup, we recorded 2 min of playing with the PLXTRM. In this time, we recorded the occurrences of an onset detection on the preamp signal, together with the timings of the onset prediction using the reservoir, and the onset detection using only a threshold on the signal received from the piezo element. A comparison of the timings of these onset events can be seen in Figure 7. In total, there were about 300 onsets detected using this approach.

Using the reservoir computing paradigm for sensor fusion outperforms predictions using the piezo sensor. To dig further into this, we compare the precision and recall of both methods and look whether the timing of these methods is accurate.

Therefore, we define predictions which came more than 100 ms before an actual onset as a false positive, predictions between 100 and 75 ms as too soon, predictions between 75 and 25 ms as correct and predictions between 25 and 0 ms as too late. The results of these statistics can be seen in Table 1. Vice versa, we look at the number of actual onsets, which were preceded by a correct prediction (both too soon, too late and correct) and compare this to the number of onsets which were not preceded by a prediction. These results can be found in Table 2.

Using sensor fusion with reservoir computing improves the results considerably. A first observation is that the reservoir produces more timely predictions. Using only the piezo element predictions are more often too late.

A second observation is that both the piezo method and the reservoir have a good recall (of about 95%), but that the precision the reservoir achieves is higher. We reckon that the overall low precision is also related to the accuracy of the onset detection, where plucks on the guitar did not provide enough energy to be detected by the onset detection. A situation which is indeed hard to detect using the PLXTRM.

This is a very good result given the fact that we are in essence still predicting human behavior, albeit on a small timescale. Perfect predictions will never be possible, but we have shown in this paper that human behavior is regular enough to allow for the build of instruments which track the intent of the performer.

6. Expressive musical processing

All expressive effects mentioned here have been explored through artistic explorations of PLXTRM.

6.1 ‘Prob-wah-bility’: probability wah

This consists of a 2-pole resonant low pass filter, the cut-off frequency of which is controlled by the probability level of the reservoir’s stroke prediction. This probability level indicates how high the probability is that a string will be struck in the next 50 ms. The resulting sound is similar to that of an auto-wah effect, but it differs in how it reacts to the playing. This

Table 1. In this table, you find the fraction of predictions per approach which fall under a certain category. It is apparent that the precision of the predictions is higher when using a reservoir. Also, using the reservoir for sensor fusion, we were able to obtain 50% more on time predictions compared to a simple onset prediction which uses only the piezo sensor. Note that only using the piezo sensor, the prediction is often too late and thus easier.

	False positive (>100 ms)	Too soon (100–75 ms)	On time (75–25 ms)	Too late (25–0 ms)	precision
Piezo	0.326	0.067	0.419	0.189	0.674
Reservoir	0.267	0.040	0.609	0.084	0.733

Table 2. In this table, you may find the fraction of onsets per category. So we compare the same two methods discussed in Table 1. As you may see, using this method, we are able to predict roughly 95% of the onsets, and that using reservoir computing for sensor fusion, we are even able to predict more than 80% of onsets on time. Again, the piezo detection is often too late.

	Not predicted (>100 ms)	Too soon (100–75 ms)	On time (75–25 ms)	Too late (25–0 ms)	recall
Piezo	0.044	0.031	0.567	0.358	0.956
Reservoir	0.051	0.027	0.805	0.116	0.949

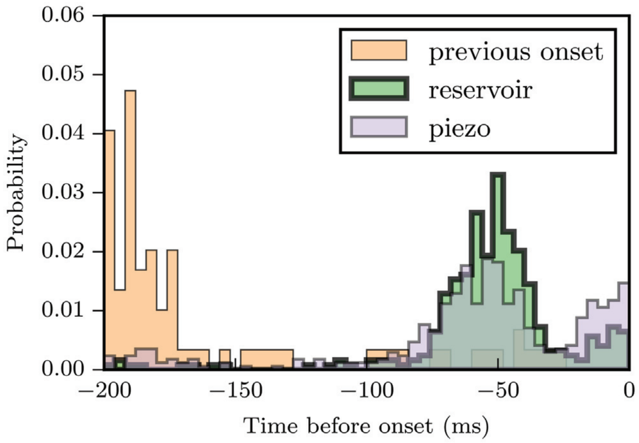

Fig. 7. Three histograms of predictions by three different methods are compared to the actual onset detection (at the zero point): onset detection by reservoir computing, by the piezo element, and the previous onsets. The reservoir onset distribution (based on sensor fusion) peeks around 50 ms before the actual onset. Compared to the predictions using the piezo only, reservoir computing onsets are more consistently detected at 50 ms before the onset (sharper and more peaks at about 50 ms). A histogram of the previous onset shows that the prediction time is of the same order of magnitude as the time between the onsets, which started around 175 ms.

effect has a more subtle relation to the playing gestures. While a traditional auto-wah effect simply responds to the amplitude envelope of played notes, this Prob-Wah-bility effect responds to the gesture of the guitar pick directly. It provides a more flexible way of interacting with the low-pass filter. The cut-off frequency can be controlled after a note has been played, but it is also congruous with played note onsets. The result is subtly different from either an auto-wah or a pedal-wah. The filter is very responsive to small picking-hand gestures and the spectral content variations in the sound follow more sophisticated patterns than with a pedal-wah. It does however take an adaptation in the mindset of the player, to adopt this method for controlling a low-pass filter.

6.2 'Note prefix'

The Note-prefix effect allows notes to be anticipated in sound before they are played. As an example, imagine the singing voice, singing a melody with words put to the melody (lyrics). For example, if the lyric of a song contains the word 'stay', the first syllable of the word ('st-') is pronounced before the musical onset of the note. A similar way of prefixing sounds to notes is achieved by PLXTRM. Through the combined use of predicted and detected onsets, sounds are introduced before played notes. The predicted onset starts a sound that continues until the detected onset stops it. This way, onsets of notes can be manipulated without disturbing the actual sounding notes. The integrity of the timing of the played notes themselves is preserved. This achieves new musical potential, where improvised single-note soloing can obtain an original personal touch not heard in conventional guitar effects. A more constant prediction time could improve playability more, especially when the attack time of the prepended sounds is short, because then new rhythmical information is added to the playing. A careful selection of suitable sounds for the prepending is crucial in this.

6.3 'Portamento'

Portamento (or 'glide') is a musical expressive device where consecutive notes are interpolated by a pitch slope. In PLXTRM, the pitch slope is not created by a continuous movement of the hand, directly mapped onto the pitch, but by the discrete events of picking two consecutive notes. When the target note, or its general location in the pitch space (lower or higher than the current pitch) can be predicted, the pitch slope can already start. Within some limitations, such as not using open strings, the string on which the target note will be played allows us to predict if the target note will be higher or lower than the source note, and thus the direction of the pitch-slope (i.e. upwards or downwards), as depicted in Figure 8(A). This approach offers significant advantages with respect to previous approached mainly based on the analysis of the played notes (no prediction), as explained in Section 7.3. While culturally the audience has come to accept the late-ness of arrival at the target notes as the defining features of the portamento effect in electronic musical instruments such as synthesizer keyboards, this new approach make the same effect more playable for the musician, and hence less distracting from the musical expression during improvisation. In spite of this, a musician using the traditional portamento approach can learn to predict the time it takes to glide to the next not, and take this into account during performance, and actually get accustomed to this method, and find the new approach to be an extra effort.

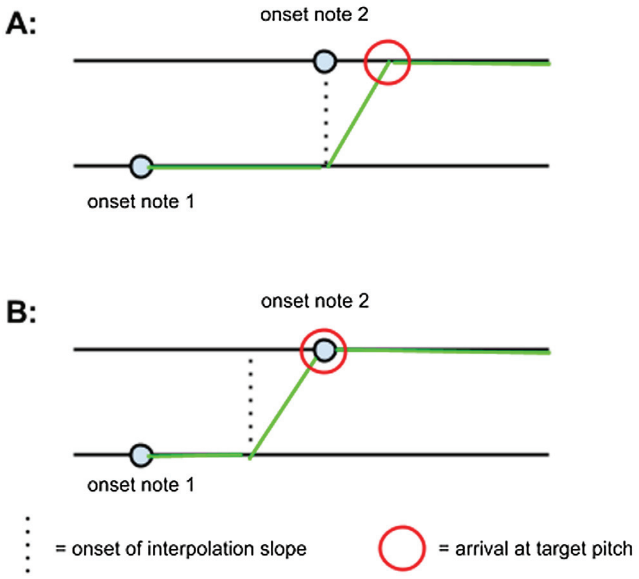

Fig. 8. This picture shows portamento start- and target points, where the slide starts at the onset of the second note (A) and before the onset of the second note (B).

6.4 ‘Hybrid guitar’

The hybrid guitar is an interactive generative guitar. The played notes of the guitar are replaced with sampled notes recorded from the same guitar. Pitch tracking is only used indirectly for this application. Instead of selecting a replacement note by tracking the currently played note, a history of recently played notes is kept. Based on the notes that have been played in the recent past, selections are made for future notes. The note selections are derived from probability tables based on the embodied mechanical properties of guitar playing. Taking standard guitar tuning as a reference, it is assumed that when two consecutive notes are played on the same string, the resulting interval between them will be at most a natural fifth, but will be most likely smaller than that. Spanning 5 frets between two notes on the same string is possible for advanced players but not often used in the lower region of the fretboard (frets 0 to 12). The distance between frets on a guitar of the most commonly used scale length of 25.5 inch (64.77 cm) ranges from 3.63474 cm between fret 0 and fret 1 to 1.0795 cm between fret 21 and fret 22 (French, 2008). This means that intervals played on the lower region of the fretboard (frets 12 to 22) need more physical distance to be spanned by the fretting hand than higher on the fretboard (frets 13 and higher), and therefore, the 5th interval on one string is less unlikely to be used on higher regions of the fretboard. It can be assumed that this interval will occur less frequently between consecutive notes played on one string. We therefore give the 5th interval the lowest non-zero probability value. More likely intervals to be played between two notes on the same string are the second and third intervals, spanning one, two, three or four frets. These intervals are given higher probability values. Repeating the same note (spanning 0 frets) is given a lower probability value, but still higher than the 5th interval. Finally, a selection of probability values is made as a function of artistic results, and can thus be adapted accordingly. When changing strings, relative transposition offsets corresponding to the strings are added. The intervals between the 6th (low E) string and the remaining 5 strings in standard tuning are 0, 5, 10, 15, 19 and 24 semitones, and are added to the above interval/probability values. Consecutive notes played on the 5th and 4th string respectively for example, will have a high probability of resulting in the following musical intervals: major 3rd (spanning 1 fret downwards), 4th (spanning 0 frets), augmented 4th (spanning one fret upwards), or a natural 5th (spanning 2 frets upwards). See Figure 9 for an example of the resulting probability tables used.

7. Discussion

7.1 Upstroke/downstroke timbre control

The difference in sound between up- and downstrokes is, although subtle, an important feature of the distinct guitar sound. In most existing sound manipulations for guitar, this aspect is not taken into account. Sound manipulation generally does not change depending on whether a note was produced by an up- or down-stroke. In the PLXTRM application, we implemented up/down stroke detection with two electrical contacts added to both sides of the guitar pick. These switches are copper surfaces that detect the unique voltage of the touched string. By measuring the voltage that is detected, it can be derived which of the 6 strings was touched. Two analog GPIO ports on the Axoloti board are used to measure these voltages, one for each of the two contacts on each side of the guitar pick, corresponding to up-strokes and down-stroke sides. A simple application with upstroke/downstroke distinction is the opening and closing of a low-pass filter.

least likely					most likely					least likely	
0.05	0.1	0.15	0.2	0.21	0.29	0.29	0.21	0.2	0.15	0.1	0.05
-5	0	-4	-3	-2	-1	1	2	3	4	0	5
-5	-4	0	-1	-3	-2	2	3	1	0	4	5

Fig. 9. Probability table of succeeding notes (normal distribution). The first row (in grey) indicates the likelihood that an interval is played (all decimal numbers add up to 1). The second and the third row, containing integer numbers, represent two examples of possible intervals sequences. The numbers represent frets span, where for example -1 indicates a half-tone down played on the same string. 5 and -5 are the maximum expected extension of the human hand on the fret-board, therefore representing the extremes of the table. In these examples the expected intervals are distributed symmetrically but it is not a requirement.

7.2 Non-picking gesture control

Picking gestures are small gestures and have relatively high velocity variability. Alternative gesture prediction can be introduced here that does not involve picking a string, but instead picking an imaginary string in the air, that is, hovering over the actual guitar strings. The X-axis is the axis from the left side to the right side of the guitar pick (see Figure 3). Gestural information from the accelerometer's X-axis has here been treated with axoloti's K-rate lowpass filter object, after which its amplitude is measured. This is the direction of motion parallel to the strings when holding the pick in playing position. The motion amplitudes in this direction are generally smaller during guitar playing, since the main activity when picking strings occurs as up-down gestures (Z-axis). Activating the control strategy described here requires the pick to be held rotated at a 90° angle around its Y-axis. When a strumming gesture is made in air (touching no strings) while holding the pick in this rotated position, the gesture is recognised as a specified non-picking control gesture. This can be used to switch specific sound processing chains on or off. A similar approach to employing non-picking gestures is used in Trail et al. (2012), where the performer can activate virtual control sliders by holding down the mallet on a vibraphone bar for a predefined time. This application features a prediction component just as PLXTRM does, however it only considers repetitive gestures.

7.3 About portamento

In most guitar effects systems, when a played note is detected, this is achieved by analysing its sound wave. In order to realise the portamento, the onset of the first note is followed by the interpolating slope in the direction of the second note, followed by the onset of the second note. The traditional implementation of portamento in electronic music instruments, such as a synthesiser, is to start the interpolating slope between the starting note and the target note at the onset of the target note. This implies that the pitch-slide will start when, musically, one might expect the pitch of the target note to be reached already. On a regular MIDI keyboard, the musical instrument doesn't receive any information about the next note that will be played until the corresponding key is already pressed. As such, the interpolating pitch slope can only be started when the target note has already been played. As a result, the target pitch is only reached with a delay corresponding to the portamento speed parameter (Figure 8(A)).

Alternatively the duration of notes can be artificially prolonged, or ultimately, the portamento effect can be applied to an external synthesised sound that is controlled by the guitar. The most salient example of portamento in the guitar tradition is the *slide guitar*. This is a traditional playing technique where the musician wears a sliding tube (or bottleneck) on one finger of the fretting hand. Instead of pressing down the strings against the frets, the sliding tube is pressed against the strings lightly, without touching the frets, and moved in parallel to the strings to play specific pitches. With this technique the slide guitar player decides when to start moving the slide tube towards the target pitch, and at what speed, based on their prediction of which gestures will result in reaching the target pitch at the desired time. A slide guitar simulation such as Pakarinen and Puputti (2008) provides an interaction mode where this effect is applied to an electronic instrument.

The approach adopted by PLXTRM, described in Section 6.3, allows the slope to start *before* the target note is actually played, thanks to prediction.

7.4 Educational potential

Augmented instruments hold a promising potential for music educational in two ways . First, they can enhance the learning of the standard way of playing the instrument (e.g. Keebler, Wiltshire, Smith, Fiore, & Bedwell, 2014; Xiao, Tome, & Ishii, 2014). The use of sensors introduces the possibility to quantitatively monitor different aspects (e.g. technical, expressive) of playing. As such, they can provide objective feedback that bypasses perceptual and interpretative difficulties that are often involved in interpersonal feedback processes such as the conventional teacher feedback (see Hoppe, Brandmeyer, Sadakata, Timmers, & Desain, 2006; Howard et al., 2004; Thorpe, 2002; Welch, Howard, Himonides, & Brereton, 2005). Next to stimulating the learner's self-monitoring abilities, this kind of feedback may contribute to self-regulation skills and to the learner's autonomy.

PLXTRM addresses this potential by providing feedback on the use of upstroke vs. downstroke picking. Developing awareness of different uses of picking is an essential part of the learning process, addressing technical, interpretative and expressive, and stylistic aspects of guitar playing.

In Jazz improvisation, for example, we can distinguish three main picking strategies in practical use: alternate picking, economy picking and sweep picking. Alternate picking is about the strict alternation of upstrokes and downstrokes, which can monitored in PLXTRM. A potential use of PLXTRM is to enlarge the sonic difference between up- and downstrokes, so that the learner automatically becomes more aware of picking patterns. Trials have also been done with visual cues in the form of colored stage lighting that changes according to up-or down-strokes. (e.g. red light for up-stroke, blue light for down-stroke). Such an approach may stimulate a more learner-centred approach in which the learner can explore and experiment with the various effects of his picking. In this way, the use of PLXTRM may complement more traditional approaches in which picking may be taught by putting forward the 'correct' model, and by instructing the learner to execute pick strokes according to this model. Additionally, the PLXTRM approach may - through the increased awareness - facilitate the internalisation of the picking patterns and as such contribute to playing technique.

Second, augmented instruments can extend the conventional ways of playing the instrument. They often create an intermediary playground that allows exploring and experimenting with action-sound relationships and, as such, possibly increase consciousness of this relationship through questioning the fixed action-response mechanism of acoustic musical instruments (Craenen, 2012)). For example, PLXTRM's musical application Hybrid Guitar offers the possibility to establish such an explorative space by mapping gestural aspects of playing a new sonic outcome, using samples.

Furthermore the note prefixes and the non-picking gesture control can be used to increase awareness of gestural aspects of playing, and to stimulate the deliberate use of expressive and communicative gestures through sonic rewards. Both allow to couple additional sonic information to the player's actions. By exploiting the subtle relationship between sonic outcome and playing gestures, it may increase awareness of the role of different gestures involved in playing a musical instrument (sound-producing, sound-facilitating, sound-accompanying and communicative gestures) as related to musical expressiveness (Godøy & Leman, 2010). In addition, this may increase awareness about the efficiency of one's picking gestures, revealing aspects of used energy and distance traveled by the pick. In this way, PLXTRM may also contribute to developing the techniques of economy picking and sweep picking. When learning any of these specific picking techniques, the student is required to pay attention to many things outside of the mere picking patterns themselves. The sequences of up and down strokes and string selections are only one aspect of the learning process in learning these techniques. When the student can get feedback about the accuracy with which a picking pattern was executed, during or after practice, more cognitive resources can be freed to direct attention towards other aspects such as hand positioning, pick-slanting or indeed, musicality.

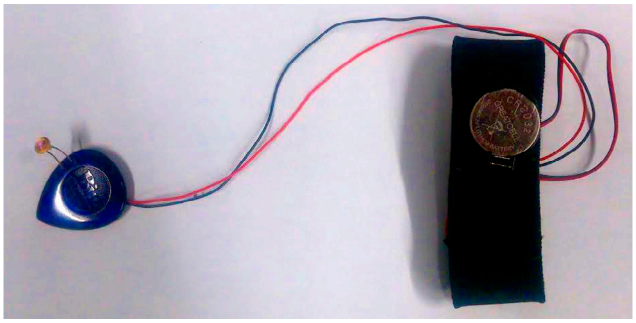
Fig. 10. guitar pick with infrared (IR) LED and wrist-band battery, for wii camera version.

7.5 Previously evaluated techniques

Our initial approach was to use optical tracking. We used a wiimote camera module for tracking an infrared LED on the guitar pick. The wiimote camera module can track up to four infrared light sources. The frame rate is 100 Hz and the image resolution is 1024×768 pixels. Most importantly, the tracking is done on-chip, so it doesn't require additional processing to be done on the Axoloti board, to which is was connected directly using a Serial Peripheral Interface (SPI) bus for communication.

This solution needed an IR LED to be mounted on the guitar pick. Although the tracking results were very useful for the prediction algorithm, several practical issues occurred. The IR LED needs to be pointed directly at the camera module in order to be tracked correctly. Moving the pick outside the field of view, or rotating it so that the IR LED points away from the camera, resulted in gaps in the tracking. The system was also sensitive to changing light conditions, which were prone to confuse the system.

For string identification, an alternative method was also attempted using 6 Hall-effect sensors. The sensors were placed under the strings, close to the 22nd fret of the fretboard. Continuous Hall-effect sensors as well as switching ones were used. A miniature neodymium magnet was mounted inside the guitar pick, near the tip. The main problem with this method was that the sensing range of the sensors was too small. The sensors reacted only when the guitar pick was passing near to them. The playing technique would have had to be adapted in order to stay close to the sensors during playing. Other systems have based the note-level prediction on pre-recorded high-speed camera footage (Oda, Finkelstein, & Fiebrink, 2013), making them unsuitable for real-time applications, which are key to PLXTRM instead.

8. Conclusions and future work

The concepts and their applications for the use of note-level predictive processing in live performance presented in this paper may bring some attention to an under-explored area of augmented guitar development, namely the treatment of the right-hand activity during playing. In particular, this paper proposes a new approach to this treatment, based on reservoir computing. Gestural control strategies as described in Section 7.2 can be extrapolated to a wide range of applications. This article presented PLXTRM, a system tracking picking-hand micro-gestures for real-time music applications and live performance. PLXTRM implements gesture recognition using a reservoir similar to the one described in Section 4.5 for the onset prediction. The PLXTRM system can be applied in music composition or live performance, and it has potential to provide music teachers with new tools to increase the self-perception of the student's stroke on the guitar strings by sonifying his/her movements, including the up- and down-strokes with the guitar pick.

On the technical side, adding a gyroscope to the pick will allow using pick inclination as a continuous controller, and more effective use can be made of rotational movement of the pick around its X, Y or Z-axis to control sound parameters. Hybrid Picking is a technique where the guitar pick is used in conjunction with plucking some strings with the fingers of the picking hand. With this playing technique, note onsets are detected on the audio signals that have no corresponding pick onset detected. Special actions can be connected to this polyphonic playing style. The inclination of the guitar pick is indeed considered integral to the sensorimotor control of the guitar player. Pick inclination is a parameter that is used naturally and often unconsciously when playing a normal guitar, as a means to influence the timbre of the played notes, as well as for efficiency in technically challenging musical material. Specific commands can be given to the system by finger-picked notes, and those commands can be modified depending on the detected position of the pick. An example of this idea would be the following instruction to the musician: 'Pick the 5th string with the pick, and simultaneously pluck either the first or second string with a finger.' The resulting command given to the system will be the same in both cases, but two different states or arguments will be connected to it, for example increasing or decreasing the intensity of an effect, depending on whether the musician combines the first or second string with the picked string. In this way, the control of electronic sound processing can be tightly interwoven with the playing technique of a seasoned guitar player. The musician's attention no longer needs to be divided over external controls and the guitar playing itself to, for example, increase or decrease an effects parameter, but this action becomes intrinsic to the playing itself. The authors are currently extending the system to support polyphonic playing styles, for which a guitar with electrically isolated frets is required.

Acknowledgements

The authors would like to thank Johannes Taelman for providing technical support to this project. Taelman is an independent developer and creator of the Axoloti board and community: http://www.axoloti.com/.

Funding

PLXTRM has a patent pending with the European Patent Office under the [submission number EP16166103.8]. The research leading to these results has received funding from the Agency for Innovation by Science and Technology in Flanders (IWT) and a Methusalem grant from the Flemish Government.

References

Bressan, F. (2014). The preservation and restoration of systems for automatic music performance. Canazza, S., & Rodà, A. eds. *Proceedings of the 1st International Workshop on Computer and Robotic Systems for Automatic Music Performance (SAMP14) in conjunction with the 13th International Conference on Intelligent Autonomous Systems (IAS)* (pp. 1–8). Venezia (Italy).

Craenen. (2012). Instruments for new ears. *World new music magazine, 22*:90–99.

Essl, G., & O'modhrain, S. (2006). An enactive approach to the design of new tangible musical instruments. *Organised Sound, 11*, 285–296.

Fala, J., Keshap, A., Doerning, M., & Barbeau, J. (1991). Note sensing in m.i.d.i. guitars and the like. *US Patent 5,033,353*.

Fernando, C., & Sojakka, S. (2003). Pattern recognition in a bucket. *Advances in artificial life* (pp. 588–597). Berlin: Springer

French, R. M. (2008). *Engineering the guitar: theory and practice*. Berlin:Springer Science & Business Media.

Godøy, R. I., & Leman, M. (2010). *Musical gestures: Sound, movement, and meaning*. New York: Routledge.

Holm, P., & Williams, B. (2014). Electronic guitar pick and method. *US Patent App. 14/092,709*.

Hoppe, D., Brandmeyer, A., Sadakata, M., Timmers, R., & Desain, P. (2006). *The effect of real-time visual feedback on the training of expressive performance skills*. 9th International Conference on Music Perception and Cognition (ICMPC9) The Society for Music Perception & Cognition (SMPC) and European Society for the Cognitive Sciences of Music (ESCOM), Bologna, Italy.

Howard, D. M., Welch, G. F., Brereton, J., Himonides, E., DeCosta, M., Williams, J., & Howard, A. W. (2004). Winsingad: A real-time display for the singing studio*. *Logopedics Phonatrics Vocology, 29*, 135–144.

Keebler, J. R., Wiltshire, T. J., Smith, D. C., Fiore, S. M., & Bedwell, J. S. (2014). Shifting the paradigm of music instruction: implications of embodiment stemming from an augmented reality guitar learning system. *Frontiers in psychology, 5*:1–18.

Kessous, L., Castet, J., & Arfib, D. (2006). *'gxtar', an interface using guitar techniques*. Proceedings of the 2006 conference on New interfaces for musical expression (pp. 192–195). Paris, France: IRCAM–Centre Pompidou.

Lähdeoja, O. (2015). *An augmented guitar with active acoustics*. SMC Proceedings of the Sound and Music Computing Conference. Retrieved from http://smcnetwork.org/node/1888

Lähdeoja, O., Wanderley, M., & Malloch, J. (2009). *Instrument augmentation using ancillary gestures for subtle sonic effects*. Proceedings of the SMC 2009–6th Sound and Music Computing Conference (pp. 327–330). Porto (Portugal).

Larger, L., Soriano, M. C., Brunner, D., Appeltant, L., Gutiérrez, J. M., Pesquera, L., Mirasso, C. R., & Fischer, I. (2012). Photonic information processing beyond turing: an optoelectronic implementation of reservoir computing. *Optics Express, 20*, 3241–3249.

Larsen, J. V., Overholt, D., & Moeslund, T. B. (2013). *The actuated guitar: A platform enabling alternative interaction methods*. SMC Proceedings of the Sound and Music Computing Conference (pp. 235–238). Logos Verlag Berlin, Stockholm, Sweden.

Leman, M. (2007). *Embodied music cognition and mediation technology*. Cambridge, MA: The MIT Press.

Leman, M. (2016). *The expressive moment: how music interaction shapes empowerment*. Cambridge, MA: The MIT Press.

Lukoševičius, M. & Jaeger, H. (2009). Reservoir computing approaches to recurrent neural network training. *Computer Science Review, 3*, 127–149.

Maes, P.-J., Leman, M., Palmer, C., & Wanderley, M. M. (2014). Action-based effects on music perception. *Frontiers in Psychology, 4*, 1–14.

Maes, P.-J., Nijs, L., & Leman, M. (2015). A conceptual framework for music-based interaction systems. In R. Bader (Ed.), *Springer Handbook in Systematic Musicology*. Berlin Heidelberg: Springer.

McPherson, A. P., Jack, R. H., & Moro, G. (2016). *Action-sound latency: Are our tools fast enough?*. Proceedings of the international conference on new interfaces for musical expression (NIME 2016) (Vol. 16, pp. 2220–4806). Queensland Conservatorium Griffith University. 20–25, Brisbane (Australia).

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

Moens, B., & Leman, M. (2015). Alignment strategies for the entrainment of music and movement rhythms. *Annals of the New York Academy of Sciences, 1337*, 86–93.

Moens, B., Muller, C., van Noorden, L., Franěk, M., Celie, B., Boone, J., Bourgois, J., & Leman, M. (2014). Encouraging spontaneous synchronisation with D-Jogger, an adaptive music player that aligns movement and music. *PLoS ONE, 9*(12), 1–40.

Moens, B., van Noorden, L., Leman, M., Nordahl, R., Fontana, F., & Brewster, S. (2010). *D-jogger: A multimodal music interface for music selection based on user step frequency*. Proceedings of the Haptic and Audio Interaction Design Conference (pp. 2). Barcelona: Universidad Pompeu Fabra.

Oda, R., Finkelstein, A., & Fiebrink, R. (2013). Towards note-level prediction for networked music performance. *NIME. 2*, 94–97.

Pakarinen, J., & Puputti, T. (2008). *Slide guitar synthesizer with gestural control*. Proceedings of 8th International Conference on New Interfaces for Musical Expression (NIME08) (pp. 5-7), Genoa, Italy.

Paquot, Y., Duport, F., Smerieri, A., Dambre, J., Schrauwen, B., Haelterman, M., & Massar, S. (2012). *Optoelectronic reservoir computing*. *Scientific Reports, 2*, 1–15.

Perng, C., Smith, J., & Rossing, T. (2011). *Harpsichord sound synthesis using a physical plectrum model interfaced with the digital waveguide*. International Conference on Digital Audio Effects (DAFx-11), IRCAM – Centre Pompidou, Paris, France.

Polson, R. (1982). Digital high speed guitar synthesizer. *US Patent 4,336,734*.

Schrauwen, B., Verstraeten, D., & Van Campenhout, J. (2007). An overview of reservoir computing: Theory, applications and implementations. *Proceedings of the 15th European symposium on artificial neural networks*. Bruges, Belgium: Citeseer.

Thorpe, C. W. (2002). Visual feedback of acoustic voice features in voice training. *Esitelmä pidetty Australian kansainvälisessä puhetieteen ja-teknologian konferenssissa*. Retrieved from https://www.singandsee.com/articles/SST2002-Thorpe098.pdf

Trail, S., Dean, M., Odowichuk, G., Tavares, T. F., Driessen, P. F., Schloss, W. A., & Tzanetakis, G. (2012). Non-invasive sensing and gesture control for pitched percussion hyper-instruments using the kinect. *NIME*. Retrieved from http://vhosts.eecs.umich.edu/nime2012/Proceedings/NIME2012WebProceedings.html

Vanegas, R. (2007). The midi pick: Trigger serial data, samples, and midi from a guitar pick. *Proceedings of the 7th International Conference on New Interfaces for Musical Expression* (pp. 330–332). New York: ACM.

von dem Knesebeck, A., & Zölzer, U. (2010). Comparison of pitch trackers for real-time guitar effects. *Proceedings of the 13th International Conference on Digital Audio Effects (DAFx)* (pp. 266–269). Graz.

Welch, G. F., Howard, D. M., Himonides, E., & Brereton, J. (2005). Real-time feedback in the singing studio: An innovatory action-research project using new voice technology. *Music Education Research, 7*, 225–249.

Xiao, X., Tome, B., & Ishii, H. (2014). *Andante: Walking figures on the piano keyboard to visualize musical motion*. In B. Caramiaux, K. Tahiroglu, R. Fiebrink, & A. Tanaka (Eds.), Proceedings of the 14th International Conference on New Interfaces for Musical Expression. Goldsmiths: University of London.