Demo of a Smart Musical Instrument-based Real Time Pattern Detection System

Nishal Silva
Department of Information Engineering and Computer Science, University of Trento
Trento, Italy
nishal.silva@unitn.it

Luca Turchet
Department of Information Engineering and Computer Science, University of Trento
Trento, Italy
luca.turchet@unitn.it

Figure 1: The smart electric guitar outfitted with the MIDI Tracker.

ABSTRACT

This manuscript focuses on a real-time pattern detection system using smart musical instruments, and its importance in Internet of Musical Things (IoMusT) applications, where smart musical instruments equipped with wireless connectivity and embedded computing devices can detect musical patterns and use them as controls for various peripheral devices. The demonstration showcases a pattern detection algorithm controlled by a digital musical instrument. The algorithm is capable of identifying pre-defined patterns during live performances and using them to trigger peripheral devices and other stage equipment. The demo features two smart musical instruments, a smart guitar, and a smart keyboard, each equipped with embedded computing devices running the real-time pattern detection algorithm.

AM '24, September 18–20, 2024, Milan, Italy

ACM ISBN 979-8-4007-0968-5/24/09
https://doi.org/10.1145/3678299.3678359

CCS CONCEPTS

• **Applied computing** → **Sound and music computing**; • **Computer systems organization** → *Embedded and cyber-physical systems*; • **Information systems** → **Music retrieval**.

KEYWORDS

Smart Musical Instruments, Real-Time Pattern Recognition, Internet of Musical Things

ACM Reference Format:
Nishal Silva and Luca Turchet. 2024. Demo of a Smart Musical Instrument-based Real Time Pattern Detection System. In *Audio Mostly 2024 - Explorations in Sonic Cultures (AM '24), September 18–20, 2024, Milan, Italy*. ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/3678299.3678359

1 INTRODUCTION

This manuscript will highlight a real-time pattern detection system based on smart musical instruments. Musical pattern detection is a widely studied topic in the field of Music Information Retrieval [1, 2, 4–7], as studies have shown that humans associate the structure of music through the perception of repetition and other relationships within a composition [3, 8, 11]. Although pattern detection in offline contexts is widely studied, real-time implementations has been limited attention.

Real time pattern detection is an important aspect in Internet of Musical Things (IoMusT) applications, which fall in the intersection between the domains on music technology and the Internet of Things [10]. Such implementations are a primary part in enabling the applications centered around the so-called smart musical instruments [9]. These digital musical instruments carry wireless connectivity capabilities, along with embedded computing devices specifically designed for real-time audio processing and networking. The intelligence of such instruments can be exploited to extract musical patterns, and then re-purpose them as controls for commented peripheral devices such as smoke machines, stage lights, visuals on screens, as well as smartphones of audience members in participatory live music contexts.

Smart musical instruments, similar to Internet of Things (IoT) devices, also posses the capability to communicate with other instruments, as well as cloud computers and peripheral devices. Such capabilities may be exploited to send control messages to trigger peripheral devices and other stage equipment. These wireless communicating capabilities may also be used to transfer computationally intensive tasks (such as training of a machine learning model) to other cloud computers and servers which are much more powerful.

2 THE DEMO

The demonstration will showcase a real-time implementation of a pattern detection algorithm controlled by the output of a digital musical instrument. A list of patterns is defined by the musician prior to the performance. These patterns may be thought of as iconic musical phrases that can used as triggers to activate peripheral devices. The algorithm is able to immediately identify when a musician plays on of the said patterns on their musical instrument. Upon recognizing that a pattern has been played, a control message is sent to a digital multiplex (DMX) controller which is programmed to trigger different peripheral devices for each control message.

The demo will present the usage of two smart musical instruments: a smart guitar fitted with the Fishman Tripleplay[1] to convert the notes into MIDI (refer fig 1), and a smart keyboard. The instruments contain an embedded computing device which runs an algorithm capable of detecting if a sequence of notes played on the instrument is a match to a list of pre-defined patterns in real-time. The algorithm takes a symbolic input in the form of MIDI. The algorithm has been developed as a Virtual Studio Technology (VST) plugin, which will run on an embedded computing device running Elk Audio OS[2].

As explained earlier, the usage of the system lies in two phases: the *Preprocessing and training* and the *Performance*.

2.1 Preprocessing and training

During the preprocessing stage, the musician will record a set of musical patterns that can be used to trigger peripheral devices. These patterns will act as the ground-truths for the pattern detection algorithm. The recording is done using the embedded system itself. Upon completion of recording, the patterns are transmitted to a remote server over the internet as illustrated on figure 2.

The training of the RNN model will take place on the remote server upon successful delivery of the recorded patterns. The synthetic data for training is generated on-the-fly, and used to train the model. The trained model is then transferred to the embedded device over the internet. During the prepossessing stage, the musician assigns an Open Sound Control (OSC) message to each pattern to be used to triggers.

2.2 Performance

Upon receiving the trained model from the remote server, the system will be ready to use. The MIDI input is sampled every 30 milliseconds to create a discrete time series for the RNN inference. A sequence of MIDI notes is obtained through windowing the incoming stream. The size of the window will depend on the length of the ground-truths.

The windowed, sampled MIDI sequence is then used to obtain a classification through the trained model. The model is able to classify if the sequence belongs to one of the pattern classes, or if the non-pattern class. If a sequence is predicted to belong to one of the pattern classes, the OSC message assigned to that particular pattern is sent to the DMX controller, which is programmed to output the corresponding DMX command to trigger the desired peripheral device.

During the demo we will illustrate the two smart musical instruments. Secondly, we will record several musical patterns as ground-truths, and demonstrate how the model is trained on the remote server. Thirdly, we will showcase the usage of the system, where the recorded patterns will be used to trigger two stage lights, connected to a USB DMC controller. A description of the developed VST plugin, as well as a video demo can be found at *https://hotlicksvst.github.io*.

[1] www.fishman.com/tripleplay
[2] www.elk.audio

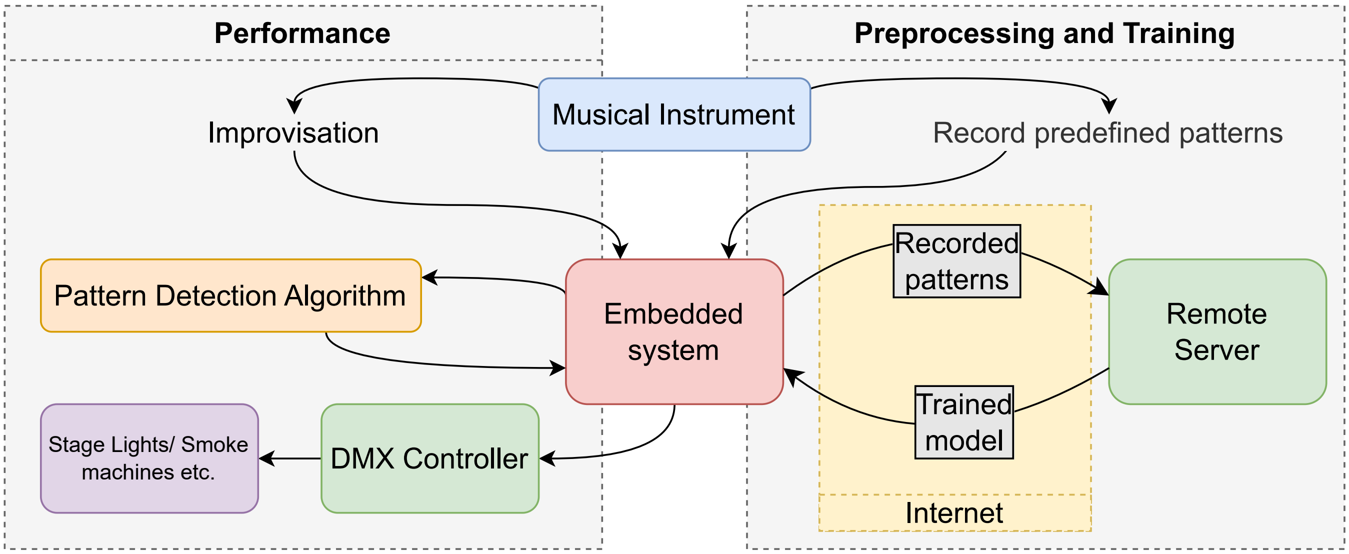

Figure 2: Block diagram of the proposed system.

REFERENCES

[1] Tom Collins, Jeremy Thurlow, Robin Laney, Alistair Willis, and Paul Garthwaite. 2010. A comparative evaluation of algorithms for discovering translational patterns in Baroque keyboard works. In *Proceedings of the International Symposium on Music Information Retrieval*.

[2] Darrell Conklin and Christina Anagnostopoulou. 2001. Representation and Discovery of Multiple Viewpoint Patterns. In *Proceedings of the 2001 International Computer Music Conference*. 479–485.

[3] Roger Dannenberg and Ning Hu. 2002. Linear time for discovering non-trivial repeating patterns in music databases. In *ISMIR 2002 Conference Proceedings: Third International Conference on Music Information Retrieval*. 63–70.

[4] Jia-Lien Hsu, C. Liu, and A. Chen. 2001. Discovering nontrivial repeating patterns in music data. *IEEE Trans. Multim.* 3 (2001), 311–325.

[5] Ian Knopke and Frauke Jürgensen. 2009. A System for Identifying Common Melodic Phrases in the Masses of Palestrina. *Journal of New Music Research* 38, 2 (2009), 171–181.

[6] Colin Meek. 2003. Automatic Thematic Extractor. *Journal of Intelligent Information Systems* 21 (2003), 9–33.

[7] David Meredith, Kjell Lemström, and Geraint A. Wiggins. 2002. Algorithms for discovering repeated patterns in multidimensional representations of polyphonic music. *Journal of New Music Research* 31, 4 (2002), 321–345.

[8] Iris Yuping Ren, Hendrik Vincent Koops, Anja Volk, and Wouter Swierstra. 2017. In Search of the Consensus Among Musical Pattern Discovery Algorithms. In *Proceedings of the 18th International Society for Music Information Retrieval Conference*. 23,27.

[9] L. Turchet. 2019. Smart Musical Instruments: vision, design principles, and future directions. *IEEE Access* 7 (2019), 8944–8963.

[10] L. Turchet, C. Fischione, G. Essl, D. Keller, and M. Barthet. 2018. Internet of Musical Things: Vision and Challenges. *IEEE Access* 6 (2018), 61994–62017.

[11] A. Volk and P. van Kranenburg. 2012. Melodic similarity among folk songs: An annotation study on similarity-based categorization in music. *Musicae Scientiae* 16, 3 (2012), 317–339. https://doi.org/10.1177/1029864912448329

Received 20 May 2024; accepted 12 July 2024