Play With Yourself - Simple Musical Accompaniment

Adar Guy
Author
adarguy@uvic.ca

Colin Malloy
Author
malloyc@uvic.ca

Hector Perez
Author
hectorp@uvic.ca

ABSTRACT

Play With Yourself is an accompaniment tool for musicians interested in making demo recordings or simply adding more texture to their solo music. The program takes a pre-recorded track of guitar or piano from the user and generates bass and drum accompaniment as a MIDI file. The MIDI file is based on extracted beat, tempo, and harmonic information. This MIDI file can then be used with virtual instruments in a digital audio workstation (DAW) to easily create a basic harmonic and rhythmic accompaniment for the original audio file.

1. INTRODUCTION

Play With Yourself is being developed as a music accompaniment system. The system is composed of two stages. The first stage is detection and extraction. The program accepts an audio file from the user and extracts tempo, beat and key information. The second stage is generative. The program outputs a bass, piano (harmonic) and drum accompaniment in MIDI format based on the extracted information acquired in the first stage. The goal of our system is to create a reasonable rhythmic and harmonic accompaniment for the input audio. The system generates a MIDI file which contains the accompaniment, and can be subsequently be used in a DAW.

This software is intended to be used by average musicians in tandem with DAW software. Play With Yourself will allow musicians to quickly and efficiently make basic demo tracks from guitar or piano recordings. The goal for this project is to create a useful tool for amateur musicians.

2. CURRENT PROGRESS

To start the project, specific libraries were tested and selected based on how closely they related to the project. As mentioned before, there are two main stages to the project - the detection, and generation stages. So far, the detection stage of the project has been addressed the most. The source code is written in the Python programming language.

For the detection stage, the feature extraction library Librosa has been used to extract tempo, onsets and beats. A separate implementation called PyMir was used for chord detection. Lastly, a MIDI library called midiutil was used to write patterns to .mid files as output to the system.

The MIDI library takes the extracted harmonic and rhythmic data as input. This information includes the duration, midi note number, and start of a note event in time. The accompaniment is represented in the MIDI file as a list of note events that provide this information.

Play With Yourself is currently designed to detect beats in 4/4 time signature and has been tested with simple guitar patterns that were recorded by a group member. There will be two parts generated for the accompaniment: one for bass guitar and one drums. For the moment, a simple bass line and drum patterns are the goal. Polyphonic lines for the accompaniment would be the next step, as well as subdividing the beats and making the drum pattern more interesting.

Due to the specialized nature of the audio input, it was necessary to record test tracks rather than use an existing data set. The benefit of creating test audio files ourselves is that we know the chord progressions and have a single, isolated instrument.

Currently, the program writes midi files as output which can then be imported into a DAW. The focus was on creating a very simple pattern that places notes generated on the first downbeat of every bar. The system does not use pre-made patterns that are just adjusted by key and tempo to create the accompaniment parts. It generates a unique accompaniment based on the harmony and timing extracted from the input audio.

The generation stage will be addressed by creating ‘rules’ for creating the patterns. These rules will be based on musical conventions in harmony. For example, tri-tones should be avoided, large skips between two consecutive notes should also be avoided, etc.

3. FUTURE PLANS

The ideal goal of the project would be to build a system with a graphical user interface, more complex patterns for harmonic and rhythmic accompaniment, as well as support for time signatures other than 4/4.

It is expected that the baseline for the project will be achieved. This entails creating a simple harmonic and rhythmic accompaniment for audio in 4/4 as a midi file that can be imported into a DAW. The generated accompaniments will be pleasant and musical. The goal in this case is to create some user controllable parameters for aspects such as onset detection sensitivity and accompaniment activity level.

In the worst case, the project will be able to identify the beats and the harmonies, but will have problems generating musically appropriate and accurate accompaniment. Probable errors include timing and incorrect chord detection.

4. TOOLS AND RESOURCES

- Pymir - jsawruk
  This is python library for music information retrieval.
- Librosa - Dan Ellis and other members at LabROSA.
  This is a Python Library for Audio and Music analysis.
- MIDIUtil - 1.1.3 : Open source object oriented library that allows the creation of multi track MIDI files.

5. LITERATURE AND PRIOR WORK

There are several previous solutions for key estimation and rhythm detection. Some of these solutions provide specific terms for implementation.

Chroma vectors are commonly used for key estimation [4, 9]. In chroma vector analysis, a high resolution fourier transform is performed on a signal and the resultant frequency spectrum is 'folded' into a 12 pitch histogram. This allows the identification of different chords.

There are many methods for rhythmic analysis. Goto and Muraoka developed two different systems that work depending on whether the input signal is drumless or not [13-14]. One system was designed for MIDI while the other dealt with acoustic audio signals. The acoustic signal approach uses a complex multiple hypothesis approach to determine the best candidate rhythm.

Other methods for beat detection use autocorrelation, binary trees or trellis trees (related to hidden Markov models). Timing nets (a form of neural network) are used in a self adjusting beat detector by R. Harper and Ed Jernigan [19]. Nodes (which represent hypothesis of where the next beat is) that are more likely to be correct predictions for the beat are chosen as the current beat for the audio signal. The nodes take in 'spike trains' and try to find coincidences between their train and the one found in the audio signal.

An MIT research project called Machine Rhythm is mentioned across much of the literature [5]. This system performs polyphonic midi rhythm detection.

E. D. Scheirer developed another rhythm detection method for acoustic multi-timbral audio [7]. Scheirer's approach uses resonators to try to achieve phase-locking with the signal. The signal's frequencies are divided into subbands and then phase-locking is attempted. The results are then combined and the tempo of the signal is extracted.

Peter Desain and Henkjan Honing developed a method using expectancy curves [18]. An overall expectancy curve for a rhythm is generated with local maxima indicating onsets of beats. Curves are generated for equal time length sections of an audio signal and then multiplied together to generate a curve that represents an average for the audio signal.

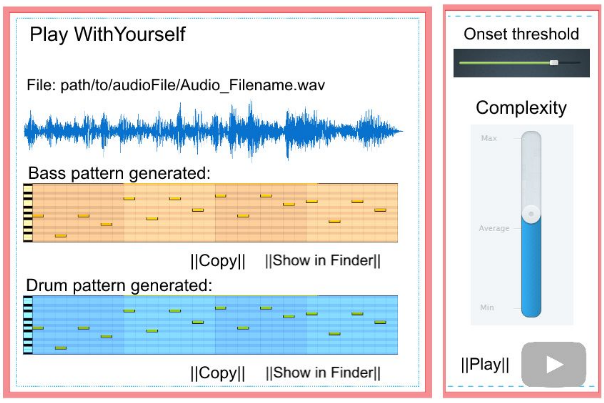
6. PLAY WITH YOURSELF IN A DAW

Play With Yourself would work well in a DAW environment but such an implementation is outside the scope of the current project.

As an ideally realized plugin version, Play With Yourself would be a plugin for a virtual instrument track that includes an analysis component and an optional sample player component. The user would select a portion of an audio track, have Play With Yourself analyze it and generate an accompaniment pattern. The accompaniment could then be customized in terms of style, busyness level, instrument sounds, etc.

The midi track generated by Play With Yourself could also be exported from the plugin and used with another virtual instrument to allow more flexible sound options.

Here is a potential representation of how the plugin might look in a DAW:



7. REFERENCES

[1] A. Driesse: "Real-time Tempo Tracking Using Rules to Analyse Rhythmic Qualities" *Proceedings of the 1991 International Computer Music Conference, 578 - 581.*

[2] A. Shenoy, and Y. Wang: "Key, Chord, and Rhythm Tracking of Popular Music Recordings" *MIT Press Journals, Computer Music Journal.*

[3] A.T. Cemgil: "Polyphonic Pitch Identification and Bayesian Inference" *International Computer Music Conference.*

[4] C. Harte, and M. Sandler: "Chromagram: Automatic chord identification using a quantised chromagram," *In proceedings of AES 118th Convention, Barcelona, 2005.*

[5] D. Rosenthal: "Machine Rhythm: Computer Emulation of Human Rhythm Perception" *MIT Media Lab, Ph.D Thesis Paper, (1992)*

[6] D. Rosenthal, M. Goto, and Y. Muraoka: "Rhythm Tracking Using Multiple Hypotheses" *Proc. of the 1994 International Computer Music Conference, International Computer Music Association, San Francisco, 1994, 85 - 87.*

[7] E. Scheirer: "Tempo and Beat Analysis of Acoustic Musical Signals" *Journal of the Acoustic Society of America, 588 - 601*

[8] G. Loy: "Musicians Make a Standard: The MIDI Phenomenon" *Computer Music Journal. Vol. 9, No. 4 (Winter, 1985), 8 - 26, MIT Press.*

[9] G. Peeters: "Chromagram: Automatic chord identification using a quantised chromagram," *In proceedings of ISMIR conference, 2006, 115 - 120.*

[10] K. Abdullah-Al-Mamun, F. Sarker, and G. Muhammad: "A High Resolution Pitch Detection Algorithm Based on AMDF and ACF" *University of Dallas Press, Citing Papers.*

[11] M. Alghoniemy, and A.H. Tewfik: "Rhythm and Periodicity Detection in Polyphonic Music" *Institute of Electrical and Electronics Engineers (IEEE), MMSP 1999: 185 - 190.*

[12] M. Davy, and S.J. Godsill in Bayesian: "Bayesian Harmonic Models for Musical Signal Analysis" *Bayesian Statistics 7, University of Montreal, 2003.*

[13] M. Goto: "An Audio-based Real-time Beat Tracking System for Music With or Without Drum-Sounds" *Journal of New Music Research, Vol. 30, No. 2, July 2001*

[14] M. Goto, and Y. Muraoka: "Real-time beat tracking for drumless audio signals: Chord change detection for musical decisions" *Speech Communication 27(3-4): 311 - 335 (1999).*

[15] N. Griffith: "Modelling the acquisition and representation of musical tonality as a function of pitch-use through self-organising artificial neural networks" *University of Exeter, UK, 1993.*

[16] P.E. Allen, and R.B. Dannenberg: "Tracking Musical Beats in Real Time" *Proceedings of the 1990 International Computer Music Conference, 140-143.*

[17] P. Cariani: "Neural Timing Nets" *Neural Network, 2001d, 14, 737 - 753*

[18] P. Desain, and H. Honing: "Advanced Issues in Beat Induction Modeling: Syncopation, Tempo, and Timing" *Proceedings of the 1994 International Computer Music Conference, 92 - 94.*

[19] R. Harper, and M. Jernigan: "Self-adjusting Beat Detection in Prediction in Music" *Institute of Electrical and Electronics Engineers (IEEE), ICASSP (4) 2004: 245 - 248.*

[20] S.A. Raczynski, E. Vincent, and Shigeki Sagayama: "Advanced Issues in Beat Induction Modeling: Syncopation, Tempo, and Timing" *Proceedings of the 1994 International Computer Music Conference, 92 - 94.*