Machine Listening Eurorack Module

Christopher Latina
Center for Music Technology, Georgia Institute of Technology

Georgia Tech | Center for Music Technology

Introduction

Machine Listening Eurorack Module is a eurorack synthesizer utility to extract instantaneous features from an incoming audio signal. Instantaneous features can be mapped to synthesis parameters for musical generation or adaptive effects.

The module is built from an open source PCB called Terminal Tedium which acts as a codec and controller for a Raspberry Pi. This allows both electronic and haptic control of functionality like thresholds, volume, and filter cutoff simultaneously.

It also uses open source software libraries including WiringPi, PortAudio and KissFFT. A real-time feature extraction library written in C++ was developed, deployed and released as an open source project.

Implementation

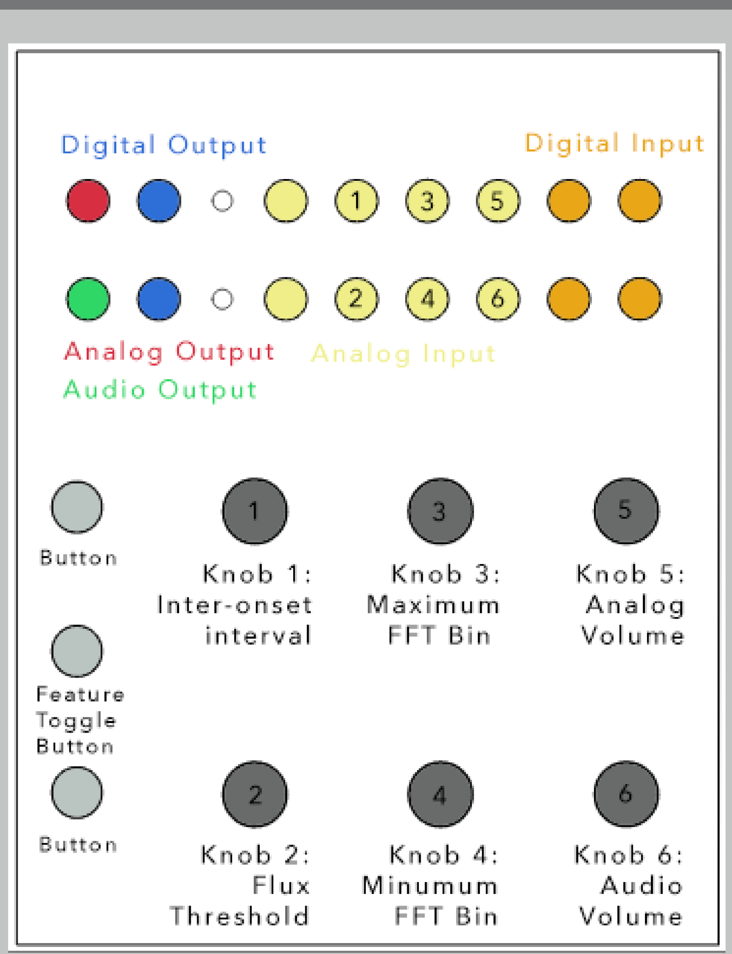

Figure 1: Interface layout

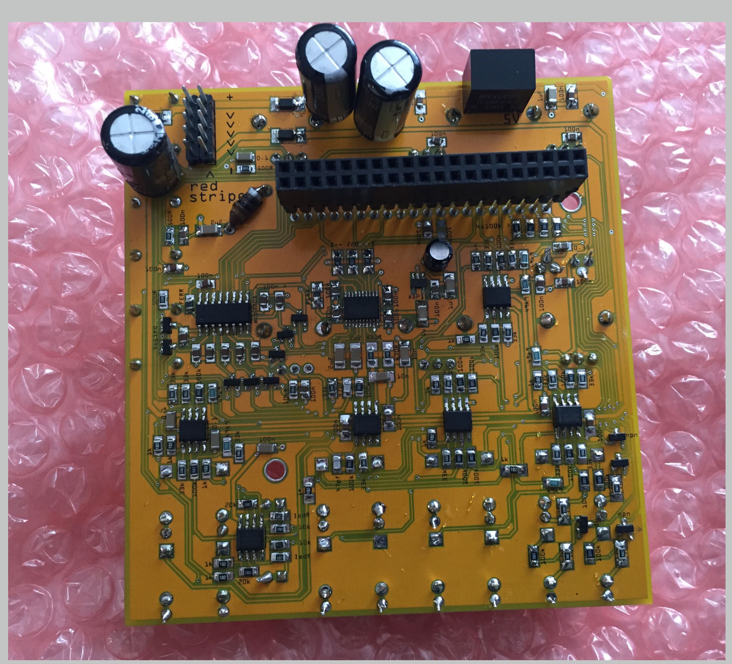

Figure 2: Terminal Tedium's PCB

The build contains a USB audio interface for audio input, 3 tact switches, 6 potentiometer knobs, 8 analog inputs (6 of which attenuate the respective knobs), 4 digital CV trigger inputs, 2 digital CV trigger outputs, and stereo audio outputs (32-bit)

1. controls the inter-onset interval ranging between 4 and 400 milliseconds.
2. controls the onset threshold (spectral flux level) ranging between 0 and 8.0
3. controls the maximum FFT bin to analyze (a low pass filter for feature detection).
4. controls the minimum FFT bin to analyze (a high pass filter for feature detection).
5. controls the volume of the feature audio / CV output.
6. controls the volume of the dry audio throughput.

Communication between Hardware and Software

- Raspberry Pi Model B+ contains 40 General Purpose Input / Output (GPIO) pins for communication hardware.
- The open source WiringPi library was used to route control voltages from an MCP3208 12-bit ADC.

The Eurorack Standard

- Terminal Tedium's build follows the Doepfer A-100 modular synthesizer standard, with the Pi powered by the module through GPIO.
- The Doepfer power specification [1] calls for +/-12V, +5V and ground rails.
- Control voltages in eurorack typically follow the 0–5V or the +/-5V standard found with pitch controlled by 1V per octave.

References

[1] Doepfer.
Technical details a-100.
http://www.doepfer.de/a100_man/a100t_e.htm.

[2] Alexander Lerch.
An introduction to audio content analysis applications in signal processing and music informatics.
Wiley, Hoboken, NJ, 2012.

[3] Simon Dixon.
Simple spectrum-based onset detection.
Music Information Retrieval Evaluation eXchange - MIREX, (x):7–10, 2006.

User Interface

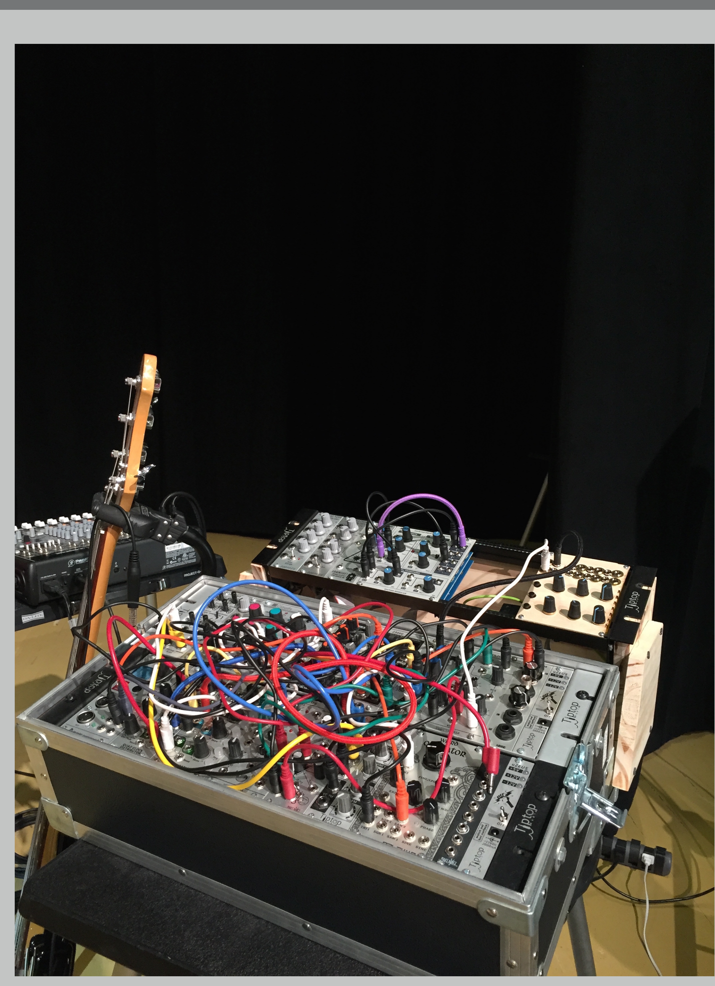

Figure 3: Machine Listening used in a performance setting

Figure 4: The front panel of the module

Features

The spectral centroid is a measure of spectral shape. Higher centroid values correspond to brighter timbre. It measures the center of gravity of the power spectrum of the STFT. This feature would be logically mapped to the bandpass filter modulated by the brightness of the incoming audio signal. [2]

$$\text{Centroid}(n) = \frac{\sum_{k=0}^{\mathcal{K}/2-1} k \cdot |X(k,n)|^2}{\sum_{k=0}^{N/2-1} |X(k,n)|^2}$$

Spectral flatness measures how tonal or noise-like a signal is. It is calculated by dividing the magnitude spectrum's geometric mean by its arithmetic mean. [2]

$$\text{Flatness}(n) = \frac{\exp\left(\frac{1}{N}\sum_{n=0}^{N-1} \ln x(n)\right)}{\frac{1}{N}\sum_{n=0}^{N-1} x(n)}$$

The spectral flux is defined as the average difference between consecutive STFT frames. It measures the amount of change of the spectral shape. Spectral flux is the technique used to detect onset triggers. The generic equation for spectral flux is as follows. [2]

$$\text{Flux}(n,\beta) = \frac{\sqrt[\beta]{\sum_{k=0}^{\mathcal{K}/2-1} \left(|X(k,n)| - |X(k,n-1)|\right)^{\beta}}}{\mathcal{K}/2}$$

Other techniques for detecting onsets exist [3], including phase deviation and complex domain calculations. I implemented a variation of spectral flux that uses half-wave rectification on the difference between each bin of the two spectra.

$$|\mathrm{X(k,n)}| - |\mathrm{X(k,n-1)}| = \begin{cases} |X(k,n)| - |X(k,n-1)|, & \text{if} \\ |X(k,n)| \geq |X(k,n-1)| \\ 0, \text{otherwise} \end{cases}$$

Conclusion

Much of this project included documentation of how to setup embedded linux devices for audio applications. Evaluation of algorithmic and hardware accuracy was done along with an evaluation of musicality. A set of recordings and compositions were presented along with a documentation of the creative work in conjunction with the development of the module. More details can be found on GitHub:

`http://github.com/chrislatina/MachineListening.`

chris.latina@gatech.edu