A Hybrid Method for Extended Percussive Gesture

Adam R. Tindale
University of Victoria
Music Intelligence and Sound Technology Interdisciplinary Centre (MISTIC)
art@uvic.ca

ABSTRACT

This paper describes a hybrid method to allow drummers to expressively utilize electronics. Commercial electronic drum hardware is made more expressive by replacing the sample playback "drum brain" with a physical modeling algorithm implemented in Max/MSP. Timbre recognition techniques identify striking implement and location as symbolic data that can be used to modify the parameters of the physical model.

Keywords

Electronic Percussion, Physical Modeling, Timbre Recognition

1. OVERVIEW

Commerical electronic drums provide a suitable amount of control for pop music that is created with samples, since these instruments are generally sample playback devices. The drummer hits a drum, the onset is detected, and a value proportional to the force of the hit is given. No information about the type of stick or the location of the drum on which it was hit is given.

Electronic drum pads come equipped with a piezo sensor that can to transduce the movement of the drumhead into an audio signal that can be connected to a standard computer audio interface. The signal is analogous to the sound that the drum makes when struck. If the drum is struck in the center, the drum will produce more low modes of vibration than if it were struck at the edge; Position tracking is given automatically through timbral variation of the input signal. Additionally, the drummer may scratch, rub, or flick the drum in any way and receive a meaningful response.

Drummers use a variety of implements on their drums in order to create a varied sound palette. Common implements included brushes, sticks, mallets, and even bare hands. For a drum interface to have wide appeal it should respond with a commensurate response to any given implement. The drum
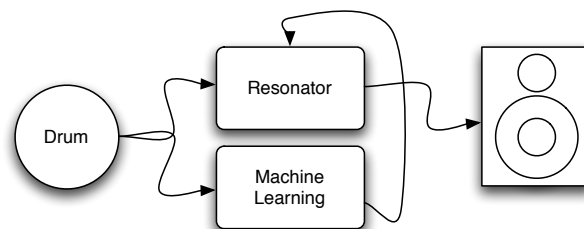
Figure 1: A diagram of the structure of the system used for the drum software.

must be passively waiting for any implement to hit it, rather than actively tracking the stick with a sensor.

A typical physical model of a drum consists of a virtual drumhead, virtual stick, and a virtual resonator. The virtual drumhead and virtual stick could be replaced with real objects and the resonator would be simulated in software. The gesture is directly translated into sound through audio processing, rather than indirect capture of a gesture via sensors.

Timbre recognition is used to determine the timbre of the drum when it is struck and infer the corresponding gesture. The timbre information is then used to modify the parameters of the physical model (see figure 1). The drum is both an instrument and a metacontroller; It can modify its own controls based on the user's input. This eliminates the need to press a button in order to change presets.

Although there have been many other drum controllers, the work of Roberto Aimi [1] is the most related. Aimi used the direct signal processing approach for a realistic interface but opted to use convolution. While many of his results sound like acoustic instruments, the ability to create a more varied sound palette is somewhat limited as well as the ability to seamlessly morph between sounds.

2. PHYSICAL MODELING

Physical modeling allows for a variable synthesis method that provides a low latency and minimal complexity. By using digital waveguide techniques [4], the software was simple to build and is very computationally efficient. The algorithm employed is a one stage digital waveguide.

The users are able to control the length of the tube (length of the delay), the dampening of the tube (feedback coefficient), and the timbre (coefficients of the filter). Presets may be stored and recalled by the user. The user may freely

Figure 2: Playing the drum with fingers, brushes and a stick.

change the character of the sound using Max/MSP's filtergraph object. While the user may not be physically informed while making these changes, often physical sounds will arise. Currently, there are presets that sound similar to bells, congas, cardboard boxes, and other presets that are somewhere in between.

3. TIMBRE RECOGNTION

When designing the type of control needed for this project, the following list of gestures and their groupings was generated:

- **Implement:** Stick, Brush, Hand
- **Stroke:** Normal, Flam, Rim
- **Radial Position:** 0-1
- **Magnitude:** 0-1

The flam stroke is produced when the user uses two sticks that hit nearly simultaneously. The radial position refers to the point on the drumhead that the drum was struck. Since there is only a single sensor in the middle, it is impossible to correlate with another source find the position in two dimensions. The implement and magnitude categories are self-explanitory.

Marsyas [6] is used to analyze the audio stream and segment it into discrete hits. This way asynchronous events can be passed when the drum is hit. Once a hit is detected, a fixed window of 4096 samples are captured to be processed. This window is sent to the feature extraction section of the software where various digital signal processing algorithms are computed. The resulting feature vector is delivered to multiple classifiers. The output of the classifiers is collected and sent via Open Sound Control.

The features used include: Root Mean Square (RMS), Temporal Centroid, Spectral Centroid, Spectral Kurtosis, Spectral Skewness, Spectral Rolloff, Spectral Flux, Subband analysis, Wavelet Coefficient Averaging and Linear Predictive Coding [3].

The features are collected into a feature vector to be classified by a guassian classifier provided in Marsyas. A gaussian classifier models the distribution of features or vectors of a particular class as a single Gaussian distribution. This distribution is characterized by the mean and covariance matrix of the training vector estimated from the training set. Gaussian classifiers are very easy to train and are fast in classification but are not particularly accurate when compared to the performance of classifiers. The classification rates drastically decrease if the distribution of the feature vector is not gaussian [2].

4. RESULTS

After some empirical tests, the timbre recognition accuracy was not adequate. This was confirmed when the result-

ing feature matrices were collected into an .arff file for Weka [7] and the actual accuracy was given (see table 1). These results show the current effectiveness of the drum and provide a goal for improvement. The mangitude metric will not be shown since that is achieved through direct signal processing. For the classification tests the two classes for Implement are stick and brush and the two classes for the stroke type are normal and rim.

	2	**3**	**4**
Implement	**86.3%**	66.1%	N/A
Stroke Type	**85.0%**	71.9%	N/A
Radial Position	**89.0%**	81.2%	73.2%

Table 1: Columns indicate the number of classes to differentiate. Rows indicate classification task.

The large analysis window is used because a previous study by the author demonstrated that larger windows increase the accuracy of the classification [5]. This latency is tolerable because the drum is always computing sound with the current audio input. The perceived effect is that triggering a patch change event will change commence on the next strike, rather than the current one.

5. CONCLUSIONS

The use of virtual resonators in conjunction with a physical device provides an intuitive interface for drummers. Timbre recognition techniques provide a suitable way of translating audio into symbolic information that can be used by the user to control the drum's controls. Direct translation of gesture through audio provides a very satisfying experience for users because of the obvious connection between gesture and result.

6. REFERENCES

[1] R. Aimi. *Percussion Using Sampled Acoustics*. PhD thesis, MIT Media Lab, 2006.

[2] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. John Wiley and Sons, Inc., New York, 2000.

[3] P. Herrera-Boyer, X. Amatriain, E. Batlle, and X. Serra. Towards instrument segmentation for music content description: a critical review of instrument classification techniques. In *Proceedings of the International Conference on Music Information Retrieval*, 2000.

[4] J. O. Smith. Waveguide filter tutorial. In *Proceedings of the International Computer Music Conference*, pages 9–16, 1987.

[5] A. R. Tindale, A. Kapur, G. Tzanetakis, and I. Fujinaga. Retrieval of percussion gestures using timbre classification techniques. In *Proceedings of the International Conference on Music Information Retrieval*, pages 541–544, 2004.

[6] G. Tzanetakis. Marsyas: A framework for audio analysis. *Organized Sound*, 4(3):169–175, 2000.

[7] I. H. Witten and E. Frank. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, San Francisco, 2005.