Indirect Acquisition of Percussion Gestures Using Timbre Recognition

Adam R. Tindale, Department of Electrical and Computer Engineering, University of Victoria, Canada
art@uvic.ca www.web.uvic.ca/~art

Ajay Kapur, Department of Electrical and Computer Engineering, University of Victoria, Canada
ajay@ece.uvic.ca www.ajaykapur.com

W. Andrew Schloss, Department of Music, University of Victoria, Canada
aschloss@finearts.uvic.ca www.finearts.uvic.ca/~aschloss

George Tzanetakis, Department of Computer Science, University of Victoria, Canada
gtzan@cs.uvic.ca www.cs.uvic.ca/~gtzan

Proceedings of the Conference on Interdisciplinary Musicology (CIM05)
Actes du Colloque interdisciplinaire de musicologie (CIM05)
Montréal (Québec) Canada, 10-12/03/2005

Abstract

There are many techniques available to capture the gestures of a performer. By utilizing digital signal processing and machine learning techniques we are able to capture, process and classify signals in real-time in order to provide data for use by musicians in a performance context. A common technique that achieves similar results is the use of sensors. The system that we describe uses data that is already available in miked environments, therefore not impeding or modifying the performer. The system is accurate and works in real-time and could be used in the future for a live performance. This system has been implemented with a snare drum and tabla for as demonstrations. The specific algorithms used, experiments and advantages and disadvantages of this technology will be discussed.

Introduction

Timbre can be defined as the variance of a sound when pitch, amplitude and phase remain constant. Though timbre characteristics are subtle, novice listeners can perceive differences when listening to music. Often the descriptions are analogous to visual colour qualities (eg. bright, dark, soft, warm) but these are vague (Grey 1978). This project attempts to represent timbre with numerical values in order to build a system for automatic recognition of musical gestures.

When one wishes to capture musical gestures, sensors are usually placed on the body of the performer or the instrument, in order to capture the motion of the gesture. A gesture is performed and a result occurs. The sensors capture the gesture and then the result can be modelled. We propose a system where a musical event is captured and then the gesture is inferred based upon the recognition of the timbre; a timbre-recognition based instrument.

This paper describes a system that analyzes musical signals, and based on their spectral content, can classify them into categories of gestures in real-time. The algorithms used for classification, the implementation, the various applications for the technique as well as future work are described.

The system has been implemented on different percussion instruments. By testing the system on different instruments (snare drum and tabla) and examining its performance in different contexts, a more general solution usable on many different instruments can be achieved. Currently the system is only tested with percussion instruments and was designed with percussion instruments in mind.

This paper aims to demonstrate the potential uses and problems for an instrument that would use these techniques. Real life applications will be used to demonstrate the magnitude of problems inherent with timbre-recognition based instruments. A description of the current system will be given, the different implementations will be discussed and then conclusions will be drawn based upon the feedback of the performers.

Background

There are two domains of related research that will be covered: timbre recognition and new interfaces for musical expression. This project borrows heavily from both fields of research in order to provide new tools for a performer.

The term *timbre recognition* is often used in the literature to refer to instrument classification. Instrument classification is the processing of an input sound to determine what instrument produced the sound. Currently, timbre classification research usually only focuses on one instrument and classifies the subtle differences between the timbres that the instrument can produce. Timbre recognition has also been explored by psychologists, for example (McAdams, Winsberg, Donnadieu, De Soete, and Krimphoff 1995), which helps to determine the accuracy of an automatic system with human performance.

Instrument classification of percussion instruments is a popular field of research because of its application to automatic transcription. Many researchers have explored this topic in significant depth, some have focused on percussion and transcribing the instruments of the drumset. Two examples of percussion instrument recognition are presented here as examples as well as the work of Caroline Traube, who has done significant work on timbre recognition of guitar tones.

Masataka Goto's beat tracking systems have application in drum transcription models. Goto's first published project (Goto 1994) demonstrates a transcription for pop music where he works under the assumption that the bass drum is often on beat one and three while the snare drum is often on beat two and four. His system included methods for recognizing snare drum and bass drum signals and then deriving a temp map by which he could perform his beat tracking.

Perfecto Herrera has also developed methods of recognizing and classifying percussion instruments for the purposes of integration into the MPEG-7 specification. He has investigated techniques that apply to drum transcription is a survey of the major classification strategies for audio content analysis (Herrera, Amatriain, Batlle, and Serra 2000). It was concluded that appropriate classification strategies need to be used to the task, and that they should be optimized for maximum efficiency and accuracy. These observations were used to create a system that can detect the presence of percussion in polyphonic streams (Sandvold, Gouyon and Herrera 2004).

Caroline Traube has created a system that is able to estimate the plucking point on the guitar based upon the resulting timbre of the stroke (Traube, Depalle, and Wanderley 2003; Traube and Depalle 2003). The work draws heavily on physics and signal processing but also includes perceptual tests of guitarists.

The ESitar (Kapur, Lazier, Davidson, Wilson, and Cook 2004) controller combines several different families of sensing technology and signal processing methods in order to capture a wide variety of gestural input data from traditional performance. In experiments (Kapur, Tzanetakis and Driessen 2004) motivated by Caroline Traube's work, sensor readings are used to train a regression model that eventually is used to replace the sensor with a virtual sensor based on the audio data. This method has a common final goal as work presented by Caroline Traube, but streamlines the human-based marking of training data with microcontroller induced sensor data.

Human Computer Interaction for Musical Performance in the context of musical performance has grown as a fusion between computer music and electronically produced music. Many different approaches have been established for creating an interface capable of the breadth of expression required for a musical instrument (Wanderley and Orio 2002). For the purposes of this paper we have provided two representative examples of significant work in this field.

Tod Machover and the Hyperinstrument Group at MIT Media Lab have created a multitude of interfaces that combine the acoustic sound of the instrument with real-time synthesis controlled by sensors embedded in the interface, dubbed hyperinstruments (Machover 1992). This is one of the few examples of serious study of synthesis and the acoustic sound of instruments combined that has been performed in public, most notably the hypercello performance by Yo-Yo Ma.

The Boie Radio Drum is a percussive gesture capture interface capable of tracking movement above the surface of the drum as well as momentary information on contact. Although it is able to convey X, Y, Z position while the sticks are within the range of reception, the precision is not reliable for complete reproducibility. A sampling rate of 200Hz is implemented in this system which is not nearly sufficient to capture the smooth motion of a stick at high velocity. The third authors' experience with this radio drum has spawned a significant amount of research in order to improve upon these limitations in the current instrument (Schloss and Jaffe 1993).

System Outline

The system that has been implemented to achieve real-time identification of the timbre of musical instruments is made up of three components: a microphone, software and a synthesizer. Many modern musical situations are either recorded or amplified, thus the presence of a microphone is not inhibitive for the performer. The instrument does not have to be modified with sensors and extra cabling is not needed. In fact, the signal from the microphones needs only to be routed to the recognition system while continuing their original purpose of amplification or recording.

The software is made up of three components: pre-filtering, feature extraction and classification. The original implementation was achieved in MATLAB. This software did not run in real-time but provided the basis for the current implementation. Experiments were carried out that demonstrated the accuracy of the system (Tindale, Kapur, Tzanetakis and Fujinaga 2004). Classification results upwards of 90% were achieved in many different experiments, the best of which being 99.8%.

Marsyas[1] is a framework for audio signal processing, music information retrieval and classification written in C++ (Tzanetakis 2000). Marsyas provides tools for musical signal processing, audio analysis, annotation and classification. Many different music information retrieval techniques are implemented in Marsyas, which allows for rapid prototyping of audio applications.

The software is bundled into a single executable that performs both the training and the classification. The software takes a series of examples from the desired instrument and trains a classifier of the user's choice. The trained classifier is stored in file so that the classifier can be recalled for further training or to be used to classify new input.

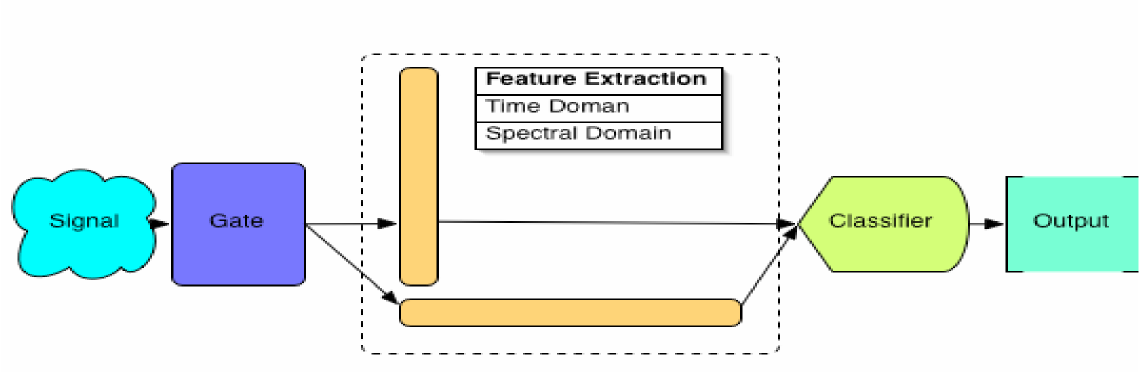

Figure 1. Outline of the software components. The feature extraction is performed on copies of the signal in the time and spectral domain and the outputs are combined, normalized and then used as input to the classifier.

[1] http://marsyas.sourceforge.net

The first component of the system is a simple amplitude threshold, or gate, that prevents the signal from being processed unless it crosses the threshold. When the signal crosses the threshold a fixed window is applied to the signal. The two strategies being investigated are a single window of 512 samples and a set of two windows of 128 samples each.

The pre-filtering section of the software is used to treat the input signal through different filtering techniques as well as to segment the signal into sections that may or may not contain an actual strike. Since the majority of the frequency content of most of the percussion instruments examined is in the lower end of the spectrum a simple lowpass filter has been employed.

If the signal is passed through the pre-filtering feature extraction is performed on that signal. 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 (Herrera, Amatriain, Batlle and Serra 2000).

The resulting feature vector is given to either train a classifier or to be evaluated by a trained classifier. Four classifiers are available to use for classification: ZeroR, Gaussian, kNN and Artificial Neural Network. More details about these classifiers can be found in (Duda, Hart, and Stork 2000).

A ZeroR classifier is used as a ground truth to compare the performace of the system. ZeroR takes a list of classes and finds the class with the greatest number of examples. The trained ZeroR classifier labels all incoming instances as the class it found to have the greatest number of examples.

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.

kNN is a classifier is a lazy learner. The training cycle of the algorithm is simply storing the feature vector. When the classifier is asked to classify new data it traverses the feature space and finds the nearest neighbour or neighbours and labels the example as the class with most number of neighbours of the same class. kNN offers a very fast training phase but can be slow in classification when the training set is large, which also consumes a great deal of memory when compared to other classifiers.

Artificial Neural Networks (ANNs) model the structure of the brain to classify data. Perceptrons are arranged in a network structure. Usually this structure is arranged in three layers: the input layer, the hidden layer and the output layer. The input layer collects the feature vector, the hidden layer passes the data from the input layer to the output layer, where the output of the system is given. Each perceptron is made up of inputs and a weight coefficient. The training of the network involves feeding the example through the network and the upon receiving output the weights of the perceptrons are adjusted so the actual output reflects the desired output. ANNs are very slow to train but are very fast for classification, they also offer the benefit of being able to solve complicated and non-linear tasks with high classification rates.

When the system is used for performance a microphone is set up to capture the signal of the instrument. The signal is routed to the computer hosting the software and the output of the recognition system is connected to synthesis software via standard protocols such as MIDI. Classification results are assigned arbitrary midi messages that are mapped in the synthesis software to meaningful parameters for sound generation or manipulation.

Case Studies

Snare Drum

The snare drum is a standard part of a drumset. What makes the snare drum interesting is the array of metal wires (snares) that rest against the bottom head of the drum. The snares vibrate sympathetically when the drum is struck. By striking the drum in different positions along the radius of the drum not only is the vibrational pattern of the membrane effected, but the excitation of snares as well. These differences in vibration manifest themselves as a change in timbre.

Seven different stroke types have been identified and classified using our system. All of the strokes were collected with the snares engaged. These strokes can be divided into three sections: Brush Strokes, Rimshot and Standard Strokes. A brush stroke is the act of striking the drum with a brush. A rimshot occurs when the membrane and the rim of the drum are struck at the same time. Standard strokes are the act of striking the membrane of the drum with a stick.

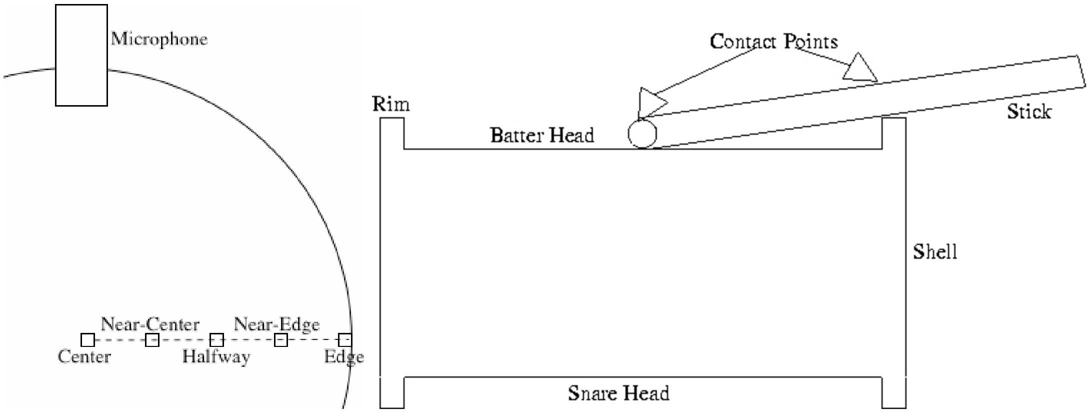

Figure 2. a) Illustration of the points along the radius of the snare drum head. b) A demonstration of the production of a rimshot

Tabla

Tabla are a pair of hand drums traditionally used to accompany North Indian classical vocal and instrumental music. The silver, larger drum (shown in Figure 3(a)) is known as the *Bayan*. The smaller wooden drum is known as the *Dahina* (shown in Figure 3(b-d)).

Influenced by research described in (Chordia 2004) our system identifies and classifies four basic tabla strokes. The *Na* stroke, shown in Figure 3(b), is executed by lightly pressing the pinky and ring finger right hand down in order to mute the sound of the drum while the index finger strikes the edge. The *Ta* stroke is executed by striking the middle and ring finger of the right hand at the center of the drum, as shown in Figure 3(c). The *Tu* stroke is executed by striking the drum with the index finger of the right hand and quickly releasing to give an open tone, as shown in Figure 3(d). The *Ga* stroke, shown in Figure 3(a), is executed by striking the *Bayan* with the middle and index fingers of the left hand. In our initial experiments we tried to recognize the *Dha* stroke, which is the *Na* and *Ga* strokes played simultaneously.

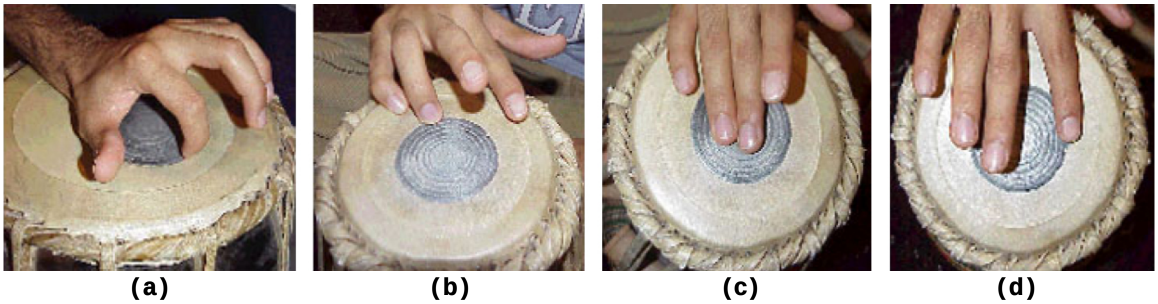

Figure 3. a) *Ga* Stroke b) *Na* Stroke c) *Ta* Stroke d) *Tu* Stroke on *Bayan* and *Dahina* of Tabla

Using the same set of soundfiles provided by Parag Chordia in his dissertation, our system achieved an overall recognition rate of 88% on 592 sound files. The confusion matrix (Figure 1) shows that *Dha* and *Na* strokes are sometimes interchanged which is explained by the fact that a *Dha* stroke in fact contains a *Na* stroke (combined with a *Ga* stroke).

a	**b**	**c**	**d**	**<- *classified as***
127	11	20	1	**a = *Dha***
4	305	4	1	**b = *Ta***
22	4	79	0	**c = *Na***
2	1	0	11	**d = *Tu***

Table 1. Confusion matrix from classification experiment with four tabla strokes.

Discussion

Timbre-recognition based instruments require that a sound be made in order to capture gesture. This leads to many design problems. The timbre produced by the instrument must be loud enough to be captured by the recognition system, yet quiet enough to not interfere with the surround musical texture.

When designing a real-time system of this nature latency is a major concern. The factors involved in classifying sounds of this nature have to do with computational complexity and the length of the buffers being computed. The original offline software used a variable sized window that was the length of the attack section of the sound. This is not practical in a real-time context, instead fixed width windows are used.

Computational complexity can be overcome with a small and efficient feature set. Our previous research has demonstrated that it is possible to attain high classification rates using only time domain features (Tindale, A. et al. 2004).

Timbre-recognition based instruments, by nature, offer interesting potential for mapping strategies. One example is the potential to use the output of the system to influence effects upon the sound captured by the microphone. This allows for the timbre of the instrument to further effect the timbre. For example, it is possible to have a delay effect on the miked signal and have the result of the output of the timbre recognition system modify the length of delay in the delay effect.

An issue with these instruments is the mapping technique to be used. Unlike other new interfaces where sensors provide a numerical reading that can be directly mapped to a parameter in the synthesis engine, the resulting data from this system is symbolic. The output does not necessitate that the symbols be organized in any standard taxonomy such as a range of values between two points. Also, when dealing with discrete classes there is possibility for problems in classification when a new piece of data is an intermediate between trained classes. One possibility is to use automatic regression techniques such as used in ESitar.

Feedback from performers was collected informally in order to help to evaluate the system. When working in controlled environments the feedback was positive. The system provided sufficient breadth of gesture recognition to allow the performer to control while still remaining flexible enough to provide great expressive potential. The biggest complaint about the system was that the data output of the system is only class; no velocity information is currently used. It was also pointed out that the system could provide the potential to control the sound after an onset. Tracking the evolution of the timbre of a sound over many windows could provide this data to a performer, much the same way aftertouch provides further data to keyboard players.

Future Research

Custom drum hardware is being constructed for use with the software system. The drum will consist of a drum resembling a timbale with a piezo and an embedded microphone. The piezo will be used to train the system in a similar manner as described in the ESitar system (Kapur et al. 2004).

A major drawback to the system is the effect of noise on the classification result. Two techniques are being investigated to combat this problem: onset detection and pre-filtering. There has been significant research into segmentation of audio streams that will be explored to improve the system. Also, a scientific study of the effect of the pre-filtering on the recognition rate is planned and the results will be absorbed into the recognition system.

Also, in an attempt to make the system more expressive for users, standard parameters will be included in future versions, such as amplitude and aftertouch-like parameters. The inclusion of aftertouch poses interesting design problems such as the need for a much more complex labelling system and corresponding examples. Since this functionality has been requested by the performers it will be investigated.

Conclusion

We have presented a new technique for capturing the subtle and expressive gestures of the expert performer. By capturing the result of the gesture by means of timbre recognition we are able to classify the timbre into a category that represents a narrow class of musical gestures. The inference of the gestures is mapped into control parameters for synthesis software. This process has been given the name of a timbre-recognition based instrument.

Acknowledgments. Thank you to Peter Driessen and Ichiro Fujinaga for their contributions to this project. Thanks to Parag Chordia for sound files for our initial Tabla experiments.

References

Chordia, P. (2004). Automatic Tabla Transcription. Ph.D. Dissertation. CCRMA – Stanford University.

Duda, R., P. Hart, and D. Stork. (2000). Pattern classification. New York: John Wiley and Sons, Inc.

Goto, M. (2001). An audio-based real-time beat tracking system for music with or without drum-sounds. *Journal of New Music Research,* 30 (2): 159–71.

Grey, J. (1977). Multidimensional perceptual scaling of musical timbres*. Journal of the Acoustical Society of America,* 61 (5): 1270–7.

Herrera, P., X. Amatriain, E. Batlle, and X. Serra. (2000). Towards instrument segmentation for music content description: A critical review of instrument classification techniques. *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*.

Kapur, A., A. Lazier, P. Davidson, R. S. Wilson, and P.R. Cook. (2004). The Electronic Sitar Controller. *In Proceedings of the International Conference on New Interfaces for Musical Expression (NIME).*

Kapur, A., G. Tzanetakis and P. Driessen. (2004). Audio-Based Gesture Extraction on the ESitar Controller. *In Proceedings of the International Conference on Digital Audio Effects (DAFX).*

Machover, T. (1992). Hyperinstruments: A progress report 1987-1991. *Media lab document*, MIT.

McAdams, S., S. Winsberg, S. Donnadieu, G. De Soete, and J. Krimphoff. (1995). Perceptual scaling of synthesized musical timbres: Common dimensions, specificities and latent subject classes. *Psychological Research,* 58: 177-92.

Sandvold V., F. Gouyon and P. Herrera. (2004). Percussion classification in polyphonic audio recordings using localized sound models. *Proceedings of the International Conference on Music Information Retrieval (ISMIR).*

Schloss, A. and D. Jaffe. (1993). Intelligent music instruments: The future of musical performance or the demise of the performer? *In Proceedings Interface 1993 (Journal for New Music Research)* 183-93.

Tindale, A., A. Kapur, G. Tzanetakis and I. Fujinaga. (2004). Retrieval of Percussive Gestures using Timbre Classification Techniques. *Proceedings of the Internaltional Conference on Music Information Retrieval (ISMIR).*

Traube, C., and P. Depalle. (2003). Deriving the Plucking Point Location along a Guitar String from the Least-square Estimation of a Comb Filter Delay. *Proceedings of the Canadian Conference on Electrical and Computer Engineering:* 2001- 4.

Traube, C., P. Depalle, and M. Wanderley. (2003). Indirect acquisition of instrumental gestures based on signal, physical and perceptual information*. Proceedings of the Conference on New Musical Interfaces for Musical Expression*: 42–7.

Tzanetakis, G. (2000). Marsyas: A framework for audio analysis. *Organized Sound*, 4 (3): 169-75.

Wanderley, M. and N. Orio. (2002). Evaluation of input devices for musical expression: Borrowing tools from HCI. *Computer Music Journal*, 26(3): 62-76.