DRUM SOUND ANALYSIS FOR THE MANIPULATION OF RHYTHM IN DRUM LOOPS

*Juan P. Bello, Emmanuel Ravelli *, Mark B. Sandler*

Centre for Digital Music
Queen Mary University of London
Mile End Road, London E1 4NS
juan.bello-correa@elec.qmul.ac.uk

ABSTRACT

This paper addresses the issue of drum sound classification in the context of automatic rhythm modification of drum loops. The proposed method segments the signal using an onset detection algorithm, characterises segmented sounds using a spectral feature set, and classifies them using k-means clustering. We propose a simple taxonomy for the grouping of different instrumental sounds under a few utilitarian labels. Results demonstrate the adequacy of our proposed taxonomy while showing that our classification approach outperforms commonly-used supervised learning techniques.

1. INTRODUCTION

Drum loops refer to short, pre-recorded percussive riffs, which are digitally repeated, processed and edited to create the rhythmic section of a given track. In recent years the use of drum loops has become an essential component of a number of popular musical genres such as techno, hip hop, dub, drum and bass, etc. Their popularity has created a growing industry for the sale of royalty-free loop repositories and software for the creation of loop-based music (e.g. Sonic Foundry's ACID, Propellerhead's ReCycle, FL Studio, etc).

From a high-level perspective, a drum loop can be described as a function of two dimensions: sound, referring to its instrumentation and acoustic properties; and rhythm, referring to the musically-meaningful organisation of the recorded sounds along time. While there is a multiplicity of tools to modify the former, there is little that can be done to automatically modify the rhythm of a drum loop. A step on this direction is taken by *FXPansion's GURU* [1], a recently released software package which attempts to transcribe drum loops, and then use the recognised segments as sound samples in a MIDI-driven sequencer. Although new rhythms can be created by rearranging sounds with this method, the naturalness of the recorded performance is lost.

Alternatively, we propose a system for the automatic rhythmic modification of drum loops. The system is able to re-synthesise a given drum loop, called the *original* signal, using the rhythm of another, known as the *model* signal. It consists of three stages: an analysis stage, where individual sounds in both signals are sliced, tagged according to their relative position to the beat, and classified into sound categories; a transformation stage where the resulting sequences of sound classes are aligned; and a synthesis stage, where the resulting alignment is used to re-allocate events on the original signal to the position of events in the model signal. Here, we only discuss the first of these stages, justifying our strategy for the analysis and classification of percussive sounds in this context.

*E. Ravelli is now with the Laboratoire d'Acoustique Musicale, Université Pierre et Marie Curie, Paris, France (e-mail: ravelli@lam.jussieu.fr).

2. BACKGROUND AND OUR APPROACH

The issue of drum sound classification has been mostly studied in the context of Music Information Retrieval, with most relevant work concerned with the transcription of isolated drum sounds (e.g. the work by Herrera et al. [2]). However, a few studies have been concerned with the classification of sounds in a drum sequence. Gouyon et al [3] presented a system for the automatic labeling of short drum kit performances, in which instruments did not play simultaneously. The audio signal was segmented using a tatum grid, and each segment parameterised as a vector of low-level features, such that instrument sounds were characterised as clusters in the feature space. Paulus et al [4] also use a tatum grid for the segmentation of drum tracks but for the more real-life case of simultaneously occurring sounds. Their system uses N-gram Hidden Markov Models (HMM) for the labeling of synthesised drum sequences. FitzGerald [5] uses a probabilistic approach to source separation for the pre-processing of the signals, thus separating simultaneously occurring sounds. In his approach an onset detection algorithm is used to slice the signal into strokes. A similar slicing approach is used by Gillet et al [6]. In their work, a number of classifiers, including HMM and Support Vector Machines (SVM), are used to recognise the instrumental sound corresponding to each sliced stroke.

In our approach we take ideas from previous methods but combine them in novel ways: we slice the signal according to onset positions and estimate a tatum-grid to label sounds according to their relative position to the beat of the loop (Section 3). We then use the standard machine learning approach to drum sound classification based on a set of low-level features calculated after the time of the onset (Sec. 5). However, as opposed to previous approaches that attempt full transcription, we classify according to a simple and novel taxonomy which is better suited to our ultimate task of rhythmic modification of drum loops (see Sec. 4).

3. ONSET DETECTION AND TEMPO ANALYSIS

A drum loop can be seen as a succession of events, each corresponding to one or many drum instrument sounds. Thus, a logical first analysis step is to slice the signal into temporal segments by detecting the beginning of each event with an onset detection algorithm. In our implementation we use the method described in [7]. This approach is based on a subband decomposition scheme using a 4-channel Conjugate Quadrature Filter (CQF), followed by the computation of a complex-domain spectral difference function on each subband. Peak picking is performed on the first derivative of the detection function as it has been shown in [8] that it improves the localisation of onsets, an issue that is of great importance for the

analysis of drum loops. For more details on the implementation and the advantages of the used algorithm refer to [7, 8].
However, in the context of the rhythmic pattern described by a drum loop, not all events have the same relevance. Indeed, events that occur at the beat, or tactus, play a greater role as they define the rhythmic periodicity of the loop. Thus, if a rhythmically-meaningful transformation is to be performed in our original loop, we need the ability to identify beat events. To do so we use a simplified version of the algorithm in [9] to estimate the tempo of the loop. We assume this tempo to be constant, which is reasonable for short drum riffs.
The sequence of onsets in a drum signal has a periodicity corresponding to the tempo of the loop. Consequently we can estimate the tempo by looking for strong periodicities in its onset detection function (DF). Periodicities in the detection function can be observed as peaks in the unbiased autocorrelation function of DF, which can be calculated as:

$$r_{DF}(l) = \frac{1}{N-l} \sum_{n=0}^{N-1} DF(n)DF(n-l), l = 0, .., N-1 \quad (1)$$

However, selecting one peak from r_{DF} results in poor resolution. For greater accuracy, we can instead average four related lag observations from it. An efficient means to extract a number of arithmetically related events from a signal (in this case rhythmically related lags from r_{DF}) is to use a matrix-based comb filter approach. However, equally weighting for all lags leads to greater variability on the detected beat periods, allowing lags of just a single sample to be considered in the estimation. To avoid this, we can use a lag weighting function such as the Rayleigh distribution. Using a weighted-comb filterbank we can then estimate the tempo of the signal, compute a grid according to the beat period and align it to the first stroke. We therefore define beat events as those located within 30ms of the estimated beat positions. In this paper we do not make further use of the knowledge about beat events. However, in an upcoming paper discussing our overall system, we use this information to address the relative importance of event timing and sound class in the context of rhythm morphing.

4. DEFINING A TAXONOMY

After the signal has been sliced into temporal segments, we now need to assign each individual slice into a sound category. This is at the core of our approach, as it will allow us to match occurrences of similar sounds in different loops. As mentioned before, we opt for the standard machine-learning approach in which we classify sounds into one of a number of pre-defined categories. However, we first need to define a taxonomy of drum sounds which is well suited to our application.
The common approach will be to use a taxonomy of drum instruments, e.g. kick drum, snare, hi-hat, conga, etc. This is equivalent to considering the classification task as a full audio-to-score transcription of the drum loops (see [6] for an example). However, for the processing of drum loops to be effective, such a taxonomy will be required to include all possible drum sounds and their combinations, resulting on all the known difficulties regarding the selection of an appropriate training set for the classifier and the generalisation to different instances of the same sound class. Moreover, this solution is far too complex and unnecessary for our application. Original and model signals are bound to have different instrumentations, and rather than attempting an approach that describes sounds according to specific instruments, we need a description that can be fitted into more general categories of drum sounds.
What we propose instead, is a simple taxonomy where drum sounds are classified according to the distribution of their spectral energy into three classes: low, mid and high.

Low: refers to all sounds (and combinations of sounds) with energy mostly concentrated on the low frequencies, e.g. the sound of a bass-drum. The *low* occurrences form the basis of a loop and define its groove, usually providing the "anchoring" points of the bar-long rhythmic pattern. The distribution of *low* events in a bar is often similar for loops of a given music style: on the 1^{st} and 3^{rd} beat of a 4/4 rock pattern, on the 2^{nd} and 4^{th} of a reggae pattern, every beat for techno, etc.
Mid: refers to all combinations of sounds where the energy content is mostly concentrated on the mid-frequency range, e.g. snare-drum, rim-shot, claps, toms, congas, etc. The *mid* occurrences balance the sequence of low events and usually bear great responsibility for the accentuation of the rhythmic pattern. The rhythm of a loop is mostly defined by the interaction between *low* and *mid* occurrences.
High: refers to all cymbal-like sounds with energy content mostly concentrated in the high frequencies, e.g. hi-hat, ride cymbal, etc. The *high* occurrences are usually used for variations, ornamentations and simple rhythmic reference. These occurrences are usually less critical when characterising the rhythmic pattern in the loop.

This taxonomy encompasses, not only a textural, but also a functional categorisation of the sounds in a drum loop. This is related to the instrumentation of "playing styles" [10], or characteristic subgroups of events in a drum pattern that are common to many a musical genre (e.g. snare or hand-clap for events on the second and fourth quarter notes of a 4/4 pattern, hi-hat or cymbal for notes occurring on the first and third notes of triplets in a swing pattern). We expect such a simple taxonomy to facilitate the classification task, while allowing generalisation to a large number of percussive sounds, both acoustic and electronically generated. In the following sections we will demonstrate that this is indeed the case.

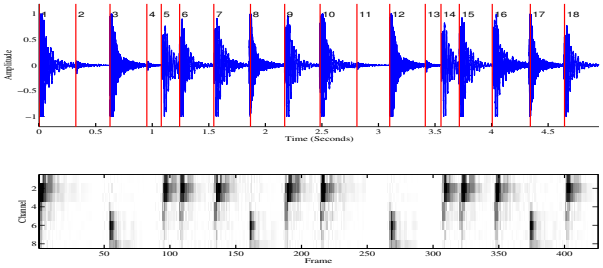

Fig. 1. The waveform of an example drum loop showing: detected and numbered onsets (top), and corresponding feature sets (bottom)

5. FEATURE SET AND CLASSIFIER

We propose a feature set that roughly characterises sounds according to their spectral magnitude shape, using the front-end implemented as part of the onset detection process. Our analysis is based on the first 8 FFT channels of the lowest subband of the CQF filterbank, corresponding to a frequency range between 0 and 350 Hz. We then construct an 8-dimensional feature vector for each sound, by taking the mean of the magnitude of these FFT channels over 100 ms after the onset (8 analysis frames). This feature set is well suited for the characterisation of our chosen categories. As can be seen in Fig.1, low occurrences (e.g. onsets 1, 5, 6 and 7) have a peaked spectrum

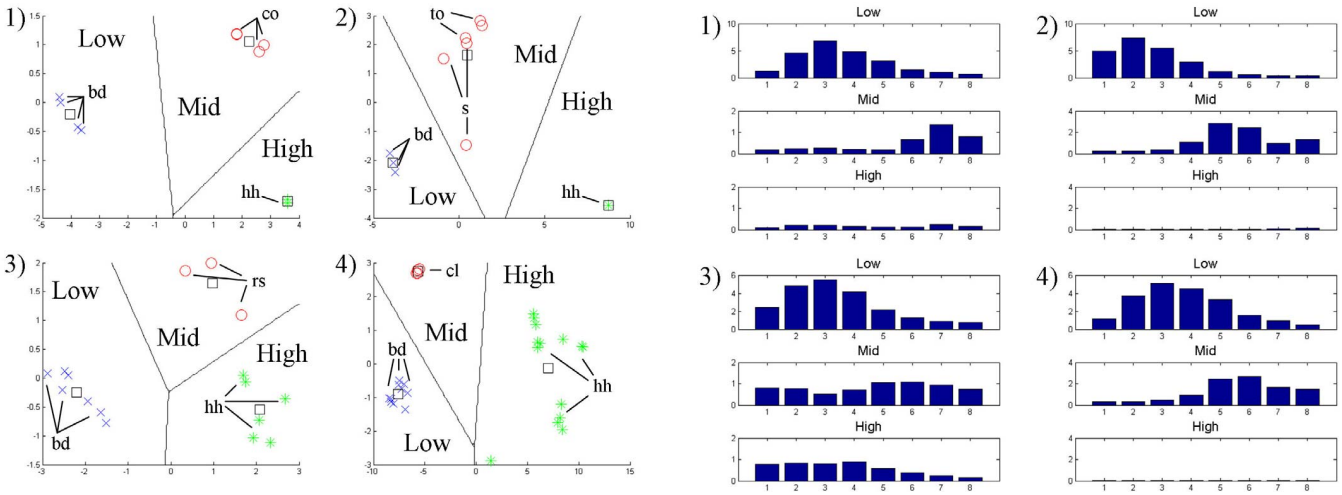

Fig. 2. (left) Clustering on four example loops: (1) an electronic-drum loop, containing bass drum (bd), hi-hat (hh) and conga (co); (2) an acoustic rock-style loop containing bass drum, snare (s), hi-hat and toms (to); (3) a hip-hop loop, containing bass drum, hi-hat and rim shot (rs); and (4) a drum loop synthesised using bass drum, clap (cl) and hi-hat. The centroid (□) and cluster boundaries are shown.

Fig. 3. (right) The spectral contents of each low, mid and high centroid for all examples in Figure 2.

concentrated on the first few channels of the FFT; mid occurrences (e.g. onsets 3, 8 and 12) have their energy spread mostly across the upper channels, while high occurrences (e.g. onsets 2, 4, 11 and 13) have very low energy, as most of their energy content is concentrated on the neglected subbands of the filterbank.

As the occurrences of each class are well separated in the feature sub-space, we can use a clustering technique for classification. Using a clustering technique eliminates the need for a target variable, or "model", necessary for supervised learning. Therefore we avoid the time-consuming creation of a large annotated training set containing a wide range of percussive sounds and their combinations. Using an unsupervised technique we create a new model for each analysed loop.

In this work we use the k-means algorithm, which divides the feature space in k clusters using an iterative calculation of each clusters' centroid (in our case k = 3). For best results, the k-means algorithm is run several times with different random initialisations; the best clustering is defined as the one with the smallest sum of point-to-centroid distances. Clusters are automatically labeled by looking at the frequency contents of each centroid.

Figure 2 shows a 2-dimensional projection of our 8-dimensional feature sub-space for four example loops: (1) an electronic-drum loop; (2) an acoustic rock-style loop; (3) a highly processed hip-hop loop; and (4) a drum loop synthesised using electronic drums (see instrumentation details on the figure's caption). The plots show the estimated centroid of each cluster (depicted as □) and the boundaries between clusters. For all examples Figure 3 depicts the spectral contents of each centroid. These examples show that our approach is able to successfully classify sounds according to our proposed taxonomy. It can be seen how different instruments (congas, toms, snare drum, rim shot and claps) are successfully clustered as mid occurrences in all experiments. This suggests that our taxonomy attains the level of abstraction needed for the rhythmic modification of loops, as we are able to map sounds with different textures to the same "functional" category. Furthermore, if we were to superimpose the plots in Figure 2, we will see that any classifier using an universal model will struggle to identify some of the mid occurrences as belonging to the same group, therefore supporting our choice of an unsupervised learning technique. The following section will be devoted to evaluate performance for a large database of drum loops.

6. EXPERIMENTAL RESULTS

In this study we use most of the database (95%) used in [6], i.e. those loops containing at least 3 instrumental classes, as our clustering algorithm cannot deal with less classes. This subset contains 300 recorded loops comprising 5400 individual strokes. The recordings are in wave format, extracted from commercial sample CDs at a sampling rate of 44.1kHz. The manual annotations on the database were mapped from their original 2^8 classes (eight instrument and all their possible combinations) to our simple taxonomy as follows: *low* contains sounds where the bass drum is predominant; *mid* groups sounds where the snare drum, clap, rim shot, tom and percussive sounds such as congas and tabla, are predominant; and *high* groups sounds where hi-hat and cymbals are played in isolation. By predominant we mean the sound that carries the highest share of the signal's energy. The database contains a multiplicity of styles including funk, jazz, rock and techno, using different drum kits, both acoustic and electronic, and in some cases processed by effects such as compression, distortion, flanger and reverberation. The length of loops varies considerably.

For our experiments we perform an empirical comparison of a few learning algorithms. Other than the proposed k-means algorithm, we use a couple of supervised learning techniques: a non-parametric learner (1-nearest neighbour or 1-NN) and a learner based on discriminant functions (a Support Vector Machine or SVM). All classifiers were implemented in Matlab. We also compare our proposed feature set to the one used in [6], composed of the mean of 13 Mel Frequency Cepstral Coefficients (MFCC), 4 spectral shape parameters and 6 band-wise frequency content parameters.

The accuracy of the supervised classification was estimated using a standard 10-fold cross-validation procedure: for each experiment,

the training set is randomly divided into 10 subsets or folds, such that we sequentially evaluate on each fold using a classifier trained on the remaining 9 folds. The cross-validation accuracy is the mean of the 10 recognition rates. For the evaluation of the unsupervised learner we operate on the whole database, with accuracy simply defined as the ratio between the number of good detections and the total number of strokes in the database.

	FEATURE SET FROM [6]	OUR FEATURES
1-NN	86.48	87.21
SVM	90.84	90.15
K-MEANS	26.20	**92.39**

Table 1. Classification accuracy for different machine learning algorithms and feature sets.

	SVM	K-MEANS
ELECTRO	82.37	**85.56**
HIP-HOP	**92.72**	90.86
ACOUSTIC HEAVY	89.53	**95.83**
ACOUSTIC LIGHT	89.74	**95.23**

Table 2. Classification accuracy for different drum-kit types

Table 1 supports our use of unsupervised learning and selection of feature set by showing that the best overall classification accuracy (92.39%) is obtained with our proposed implementation. This is convenient for our application as the 1-NN approach, which performs the worst, needs far more memory and computational power than any of the other approaches, while both the SVM and the 1-NN need a large annotated training set. Noticeably, results using our feature set are consistently high for all classifiers, showing its ability to characterise sounds according to the given taxonomy.

The performance difference between the SVM using the feature set in [6] and our implementation is less than 2%. This suggests that given a large and wide enough training set (like the database we use), this tested-and-tried supervised approach can perform similarly to our method. However, the main strength of our approach is that, by not relying on a training set, it has the ability to generalise to textures that the system has not seen before.

Table 2 shows a comparison of our two best classifiers using different "drum-kit textures". In [6] the data set was split according to the drum-kit type on each loop. Four categories were defined: *electro*, containing sounds from electronic drum kits; *hip-hop* with highly processed sounds; *heavy*, containing sounds with heavy and long reverberation; and *light*, using common acoustic sounds. We then perform a 4-fold experiment, where each sub-set of the data corresponds to a fold, such that the supervised algorithm is sequentially tested on all folds and trained on the remaining three. Apart from the case of *hip-hop* drum-kits, results consistently show how our system generalises better than the SVM for unknown drum-kits, further supporting our case for the creation of a new model for each loop. Results in the table also hint at the complexity of correctly labeling electronic or highly processed sounds.

7. CONCLUSIONS

An approach is proposed for the classification of percussive sounds in drum signals. It is part of a larger system for the automatic rhythmic modification of drum loops, to be presented in an upcoming publication. Hence the classification is not aimed at full transcription, as in previous approaches, but to a description of the contents which allows the matching of events in different signals according to their "role" in the drum sequence. It starts by slicing the signal into segments using and onset detection algorithm and calculating a tatum grid to characterise events according to their relative position to the beat. We then classify the sounds on those segments into three simple categories - low, mid and high - which are related to the spectral contents of each sound. We use a set of spectral features and a k-means clustering algorithm for the classification. Our results show that our taxonomy is well suited to map different instruments in different loops to the same functional categories, for example, the sound of snare drums, claps, congas and toms are all clustered as *mid* occurrences; while both electronic and acoustic bass drums are considered as *low* occurrences. Furthermore, experiments on a large annotated database of drum loops show that our approach outperforms the use of supervised learning techniques and a more standard feature set, while being able to generalise better to "unfamiliar" sounds. On-going experiments on our complete system for automatic rhythmic modification support the observations and claims made on this paper.

8. ACKNOWLEDGEMENTS

Thanks to O. Gillet and G. Richard for kindly providing the annotated drum loop database. This work was partially funded by the European Commission through the SIMAC project IST-FP6-507142.

9. REFERENCES

[1] FXPansion, "GURU," http://www.fxpansion.com, 2005.

[2] P. Herrera, A. Yeterian, and F. Gouyon, "Automatic classification of drum sounds: a comparison of feature selection methods and classification techniques," in *2nd Int. Conf. on Music and Artificial Intelligence*, Edinburgh, UK, 2002.

[3] F. Gouyon and P. Herrera, "Exploration of techniques for automatic labeling of audio drum tracks instruments," in *MOSART Workshop in Computer Music*, Barcelona, 2001.

[4] J.K. Paulus and A.P. Klapuri, "Conventional and periodic n-grams in the transcription of drum sequences," in *Proc. of the IEEE ICME*, Baltimore, USA, 2003.

[5] D. FitzGerald, E. Coyle, and B. Lawlor, "Sub-band independent subspace analysis for drum transcription," in *5th Conf. on Digital Audio Effects, Hamburg, Germany*, 2002.

[6] O. Gillet and G. Richard, "Automatic transcription of drum loops," in *Proc. of IEEE ICASSP, Montréal, Canada*, 2004.

[7] C. Duxbury, J.P. Bello, M. Sandler, and M. Davies, "A comparison between fixed and multiresolution analysis for onset detection in musical signals," in *the 7th Conf. on Digital Audio Effects. Naples, Italy*, October 2004.

[8] E. Ravelli, M. Sandler, and J. Bello, "Fast implementation for nonlinear time-scaling of stereo audio," in *Proc. of 8th Int. Conf. on Digital Audio Effects (DAFX05)*, September 2005.

[9] M. E. P. Davies and M. D. Plumbley, "Causal tempo tracking of audio," in *ISMIR-04, Barcelona, Spain*, October 2004.

[10] C. Uhle and C. Dittmar, "Drum pattern based genre classification of popular music," in *Proc. of the AES 25th International Conference, London, UK*, 2004.