ACOUSTIC MODELLING OF DRUM SOUNDS WITH HIDDEN MARKOV MODELS FOR MUSIC TRANSCRIPTION

Jouni Paulus

Institute of Signal Processing, Tampere University of Technology
Korkeakoulunkatu 1, Tampere, Finland
jouni.paulus@tut.fi

ABSTRACT

This paper describes two methods for applying hidden Markov models (HMMs) to acoustic modelling of drum sound events for polyphonic music transcription. The proposed methods are instrument-wise binary modelling and modelling of instrument combinations. In the first, each target instrument is modelled with a "sound" model and all target instruments share a "silence" model. Each instrument is transcribed independently from the others. In the latter method, different instrument combinations are modelled, and an additional "silence" model is created. The proposed methods are evaluated with simulations with acoustic data, and compared with two reference methods. Simulations show that combination modelling performs better than instrument-wise modelling.

1. INTRODUCTION

Drum transcription is here defined as detection and recognition of drum sounds in a given acoustic input signal. The input signal can be either a signal containing only drums, or real polyphonic music, and the output can for example be in the form of a MIDI file.

A coarse taxonomy that can be used with existing drum transcription systems divides them into *pattern recognition-based* and *separation-based* methods. The systems from the first category operate by initially finding temporal locations of possible percussive sound events, segmenting the signal at the onset locations, extracting a set of features describing the segment, and using a classification algorithm to recognise the segment. The final transcription can be obtained by combining the temporal and classification information of the segment. Systems in this category include, e.g., the systems by Gillet and Richard [3, 4], and the work of Tanghe et al.[15].

Systems from the second category utilise source separation methods to segregate each drum instrument into own streams. After the segregation, remaining task is locating the sound event onsets, and if a blind separation method was used, recognising the used instrument. The recognition step can be solved to some extent by using "a dictionary" of possible sounds already in the separation process. An example of a blind separation system is by Dittmar and Uhle [1]. Several methods, both blind and dictionary-based are presented by FitzGerald in [2]. It has been noted that when the dictionary and target signal match well, the transcription result can be good [8].

In addition to these clear categories, also some hybrid methods have been developed. They adapt the model to fit the target signal better. The adaptation can take place, e.g., by constructing signal-specific models after initial classification [13], or by adapting the rough initial models iteratively during the classification process [16].

The system presented in this paper cannot be directly categorised in either of the above classes. It adopts the hidden Markov model (HMM) widely used in the automatic speech recognition [11]. Like in many separation-based methods, the whole input signal is analysed in short frames, but instead of applying any source separation algorithm, a set of features is extracted from each frame, as is usually done in the pattern recognition-based approaches. These feature vectors are then interpreted as observations from a HMM process. The main motivation for using HMMs is that they enable modelling the sequential dependencies of consecutive observations.

To our knowledge, HMMs have not been applied in this manner for the drum transcription task earlier. They have been used for drum transcription, but the states have been attributed to events and transitions between events instead of frames covering the whole signal [3]. Hence, the earlier utilisation of HMMs resembles more musicological modelling.

The target signals that we are interested in here are polyphonic drum tracks (multiple drums playing at the same time, no other instruments) and polyphonic music (also other instruments are present).

In this paper, two different HMM-based drum transcription methods are proposed, evaluated, and compared to two reference methods. In the first, each drum instrument is transcribed independently from others with two HMMs modelling the actual sound event and signal when the instrument is not played. The second method models drum sound combinations and the situation when no drum instrument is played.

2. ANALYSIS FRONT-END

In order to enable the use of HMMs, acoustic features have to be extracted from the input signal. Here, an optional preprocessing step is described, after which the signal is divided into short frames. From these frames, a set of features is extracted, and the extracted features can be transformed into a lower-dimensional and decorrelated space to be used with the HMMs. The presented system handles only one-channel signals.

As the sound of many drum instruments, especially the idiophones (e.g., cymbals), can be considered stochastic, sinusoids+residual modelling has been proposed as a preprocessing step, e.g., [4]. It assumes that most of the non-drum instruments will be modelled with sinusoids and the drum instruments will remain in the noise residual obtained by subtracting the sinusoids from the input signal. The effect of a such preprocessing is evaluated in the simulations.

2.1. Feature extraction

After the optional preprocessing, the signal is divided into short, overlapping frames. Input signal sampling rate F_s being 44100 Hz,

frame length of 1024 samples was used (corresponding to approximately 23 ms). Frame overlap of 50% was used. Also other values were considered, but the used showed the best performance.

From each frame, a set of features are extracted. They include the first 13 mel-frequency cepstral coefficients (MFCCs). MFCCs have proved to be an efficient feature set in many audio applications, as they parametrise the rough spectral shape in a compact way. In addition to the MFCCs, their first order temporal differences (ΔMFCCs) are taken to enable simple modelling of temporal evolution of the coefficients.

Also simpler spectral features are extracted, including spectral centroid, spectral spread, spectral skewness, spectral kurtosis, spectral slope, spectral flatness, and 85% spectral roll-off point (see [10] for definitions). All of these simpler spectral features are calculated using a logarithmic frequency resolution.

Finally, the RMS value of the signal in the frame is calculated, as well as the band energy ratios of six non-overlapping octave bands (band k has upper boundary F_k given by $F_k = F_s 2^{-K+k-1}$, where K is the total number of bands).

2.2. Feature transformations

Even though the MFCCs are decorrelated from each other due to discrete cosine transformation step in their extraction procedure, similar decorrelation can not be guaranteed over the whole feature set. Instead, it is very likely that the chosen feature set has inter-feature correlations. The redundant information causes unnecessary computational load, and the covariance matrix calculated over all the features is not diagonal, which is assumed by the used HMM toolbox.

The effect of two linear, unsupervised, feature transformations methods are evaluated. They are principal component analysis (PCA) and independent component analysis (ICA). PCA aims to remove second order statistical dependencies from the data, while ICA aims to remove also higher-order dependencies. Both have been noted to be useful in phoneme recognition with HMMs [14]. In addition to the feature decorrelation, the transformations enable to reduce the feature vector dimensionality.

In the simulations, when using PCA, the number of retained dimensions was determined by choosing the ones contributing to 90% of the overall variance. With ICA, the used algorithm was requested to return 20 components. A fast, fixed-point implementation of the ICA algorithm was used [6].

3. SYSTEM ARCHITECTURES

The two alternative methods for applying HMMs in the acoustic modelling evaluated here are the instrument-wise binary modelling and the modelling of instrument combinations. In both methods, the observation distributions within HMM states are modelled with Gaussian mixture models (GMMs).

3.1. Instrument-wise modelling

In the instrument-wise modelling, each target instrument is transcribed independently from the others. For the transcription, two HMMs are used: one modelling observations during a sound event of the target instrument, and another for modelling observations when the instrument is not played. The "sound" HMM tries to model how the feature values evolve during the sound event, regardless of any other simultaneously sounding instruments. The feature values when the instrument is not played are modelled with a 1-state HMM, i.e., with a GMM. Such a simple structure is assumed sufficient because the modelled data has no distinctive temporal structure, or it is so complex and dependent on the signal that the modelling is practically impossible. Instead of constructing separate "silence" models for each instrument, all target instruments share a common model, referred as a universal background model (UBM) after [12].

3.2. Combination modelling

The second proposed method models the combinations of simultaneously occurring drum instruments, i.e., if there are kick drum (B) and snare drum (S) in the drum kit, then the modelled combinations are B, S, and B+S. As with the instrument-wise modelling, a background model is constructed to model the features when none of the combinations is active.

A drawback of modelling combinations is that their number increases rapidly as a function of the possible instruments. However, in practice a small portion of these combinations contribute the majority of all combination occurrences [9, 3]. This observation can be used to choose a subset of combinations to be modelled.

3.3. Training the models

The expectation maximisation algorithm was used to learn the parameters of GMMs and the state transition probabilities in both HMM architectures [11]. In simulations, it was noted that five HMM states for modelling the sound event produced the best result, both in combination and instrument-wise modelling. The number of components in GMMs was set to three for sound event modelling, and to seven for silence modelling. Also these values were determined with simulations.

Segmentation of the training events was done using a fixed-length segment after each manually annotated event start. Drums as freely-decaying instruments do not have a clearly-defined duration or offset time, so the used segment length was determined by analysing -40 dB decay time from a large set of individual drum sound samples. The average decay time varied from 130 ms of kick drum to 420 ms of hi-hats. Based on these results and some experiments, the maximum segment length was set to 180 ms. If another sound event of the same type sets on earlier, the segment length is reduced to match the interval between the events. It might be beneficial to determine the segment length based on the sound event itself, but the sound event length determination is a difficult task due to co-occurring instruments, both other drums and melodic.

Each of the instrument-wise and combination models are trained with features related to the occurrences of the modelled event within the training data. E.g., with instrument-wise modelling and training a model for the kick drum, all the occurrences of kick drum events in the training data were used to determine the model parameters. The background models are trained using the signal segments that are not used to train any of the other models. This approach provided the best result. The other alternatives for background model training would be to use only the frames used to train all the sound models (similar to the GMM-UBM training for speaker verification in [12]), or to use all frames from the training data, both including drums and non-drums. The method of using only the frames from sound events performed worst of the three.

3.4. Using the models

Transcription with the models is straightforward: features are extracted as was done in the training phase, feature vectors are normalised and possibly transformed using the parameters calculated

in the training phase, HMM state observation likelihoods are estimated from GMMs, and finally the optimal path through models is decoded. The difference between the evaluated architectures is illustrated in Figure 1. The top panel illustrates the usage of instrument-wise models, while the lower panel illustrates the combination modelling.

With instrument-wise modelling, the observation likelihoods of the UBM and the model of instrument to be transcribed are combined and the token passing algorithm [17] is used. The result is a path describing whether the instrument is playing in a certain time frame or not. This is repeated for all instruments. After all instruments have been decoded, the occurrence information is combined to yield the final transcription. The usage of the combination models is otherwise similar with the instrument-wise models, except instead of decoding each instrument separately, the observation likelihoods of all combinations are combined and only one path is sought.

The main difference is that the instrument-wise modelling lacks the knowledge about the context created by co-occurring instruments. As noted in [9], this may lead to the occurrence of very unlikely combinations in the transcription. The co-occurrence information is modelled implicitly with the combination modelling. Still, the trade-off between the computational complexity caused by the number of models, the co-occurrence information importance and the amount of data required to construct the models exists.

No musicological modelling was used, even though it has been shown to improve results [9]. This was done to keep the systems as simple as possible and the results easier to interpret. Hence, all transitions between models were set equiprobable.

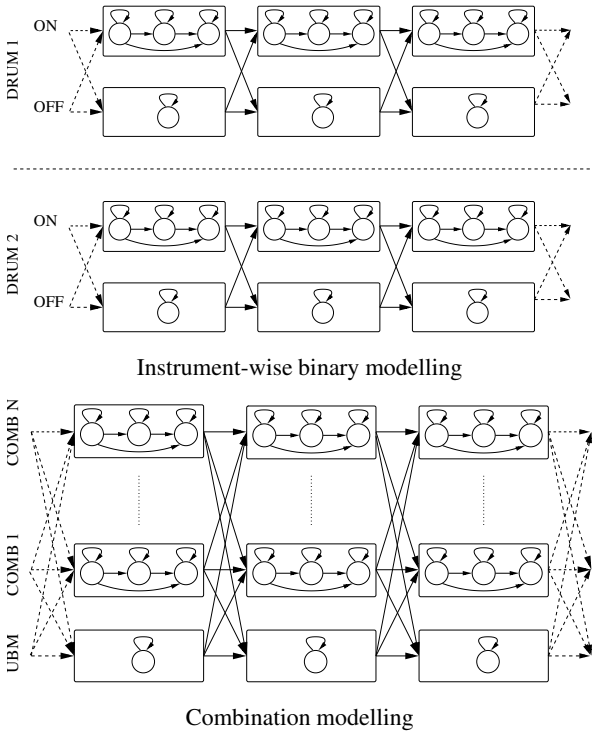

Fig. 1. Difference of the proposed HMM architectures.

4. EVALUATION

The performance of the proposed methods were evaluated with simulations with acoustic input data. The target instruments were limited to the set containing only kick drum, snare drum and hi-hats. These three are able to describe the rhythmic structure of an average pop/rock piece accurately enough for many applications. The transcription result was compared to the manually annotated ground truth. The calculation of the evaluation metrics is presented in detail in [8], allowing 30 ms deviation of transcribed events from the ground truth.

The system performance was evaluated with three sets of acoustic material. The first two sets consist of signals containing only polyphonic drum sequences, i.e., no other instruments are present. The first data set, referred as "simple drums", consists of relatively simple percussive sequences played mainly with the target drums. This set is the same as the "production grade wet mixes" in [8]. The second test set, referred as "complex drums", was recorded and annotated with the same setup as the first one, but the sequences are more complex in nature, i.e., they contain also other drum instruments and the playing has more variation. The third test set, referred as "RWC Pop", consists of the pieces from the RWC Popular Music database containing also other instruments and singing [5]. The ground truth annotations for the RWC set were generated based on the MIDI files provided along the database. However, the MIDI files were not synchronised accurately enough with the acoustic data, so temporal offset and possible tempo changes were annotated by hand.

From all pieces in the data sets, the first thirty seconds of the acoustic data was used in the evaluations. If the acoustic data was less than thirty seconds in length (as was the case in some of the pieces in the drums-only sets), the entire signal was used. 3-fold cross-validation was used, and the evaluation metrics were calculated over all folds.

The evaluation metrics were calculated both to all instruments separately and over all instruments. The used metrics were precision rate P and recall rate R, defined as

$$P = \frac{c}{s}, \qquad R = \frac{c}{g}, \tag{1}$$

where c is the number of correctly transcribed events, s the number of events in the system output, and g the number of events in the ground truth annotations. From precision and recall rate, F-measure F was calculated with

$$F = \frac{(\beta^2 + 1)PR}{\beta^2 P + R}, \tag{2}$$

where β is a weighting factor (equal weighting $\beta = 1$ was used). All the used metrics have the range of $0 \ldots 1$, and larger value is better.

To put the results in perspective, two other methods were taken for comparison. The first, referred as *NSF*, relies on source separation using a dictionary, and was presented originally in [8]. The second, referred as *SVM*, is an example of pattern recognition on events, described in [15]. From both comparison methods, the original authors' implementations were used, SVM implementation was from [7]. With NSF method, the onset detection thresholds were optimised automatically during simulations. With SVM method, no training was done and only the provided models were used.

The results are presented in Table 1 containing overall F-measures for transcribing kick drum, snare drum, and hi-hats from the three different material sets. Detailed evaluation results for the best performing HMM architecture for each evaluation material type are presented in Table 2. With all test sets the best performance was obtained with combination modelling along with the following operational parameters: "simple drums" (preprocessing and PCA), "com-

F-measure (%)	simple drums	complex drums	RWC Pop
binary HMM	72.7	65.8	44.6
comb. HMM	86.4	79.1	47.0
SVM	82.8	75.9	53.8
NSF	95.6	71.2	0.10

Table 1. Total average F-measures for the evaluated systems and different material sets. For HMM-based systems, the result gained with the optimal parameter values is presented.

material	metric	kick drum	snare drum	hi-hat
simple drums	P(%)	81.7	88.8	82.6
	R(%)	89.5	82.5	93.4
complex drums	P(%)	73.5	59.8	76.3
	R(%)	92.2	86.6	89.6
RWC Pop	P(%)	38.6	24.3	44.2
	R(%)	73.5	54.5	62.7

Table 2. Detailed results for the best performing HMM systems for each evaluation material set.

plex drums" (no preprocessing or feature transformations), and "RWC Pop" (preprocessing and no feature transformations).

It can be seen that the combination modelling performed considerably better than the instrument-wise modelling with both test categories consisting only of drums. The difference was reduced with the polyphonic music, in which case both methods performed badly. When considering the effect of preprocessing or PCA as feature transformation, nothing conclusive can be said. Even though small differences were observed in the performance, they were not consistent and may not be statistically significant.

Applying ICA as a feature transformation proved to have very little effect with the simple material. With the more complex material, ICA decreased the overall system performance noticeably.

Based on the results, the main problem with HMM-based approaches seems to be low precision, i.e., the systems tend to transcribe more events than there are present in the input signal. One way to handle this problem could be to use also features which would describe the accent of the signal. Other could be to integrate a musicological modelling to the system.

5. CONCLUSIONS

This paper has presented application of two HMM architectures in the acoustic modelling of drums sound events for music transcription. The proposed methods were evaluated with simulations with acoustic material and compared with two reference systems. The evaluation results suggest that combination HMMs perform clearly better than instrument-wise models. The effect of preprocessing and feature transformation was left inconclusive. The results also suggest that plain HMMs without any musicological modelling may not be the optimal approach to the transcription problem.

6. REFERENCES

[1] C. Dittmar and C. Uhle. Further steps towards drum transcription of polyphonic music. In *Proc. of 116th Audio Engineering Society Convention*, Berlin, Germany, May 2004.

[2] D. FitzGerald. *Automatic Drum Transcription and Source Separation*. PhD thesis, Conservatory of Music and Drama, Dublin Institute of Technology, Dublin, Ireland, 2004.

[3] O. Gillet and G. Richard. Automatic transcription of drum loops. In *Proc. of IEEE International Conference on Acoustics, Speech, and Signal Processing*, Montreal, Canada, May 2004.

[4] O. Gillet and G. Richard. Drum track transcription of polyphonic music using noise subspace projection. In *Proc. of 6th International Conference on Music Information Retrieval*, pages 92–99, London, UK, Sept. 2005.

[5] M. Goto, H. Hashiguchi, T. Nishimura, and R. Oka. RWC music database: Popular, classical, and jazz music databases. In *Proc. of 3rd International Conference on Music Information Retrieval*, pages 287–288, Paris, France, Oct. 2002.

[6] J. Hurri, H. Gävert, J. Särelä, and A. Hyvärinen. The FastICA package for MATLAB, 1998. `http://www.cis.hut.fi/projects/ica/fastica/`.

[7] MAMI. Musical audio-mining, drum detection console applications[1], 2005. `http://www.ipem.ugent.be/MAMI/`.

[8] J. Paulus and T. Virtanen. Drum transcription with non-negative spectrogram factorisation. In *Proc. of 13th European Signal Processing Conference*, Antalya, Turkey, Sept. 2005.

[9] J. K. Paulus and A. P. Klapuri. Conventional and periodic N-grams in the transcription of drum sequences. In *Proc. of IEEE International Conference on Multimedia and Expo*, volume 2, pages 737–740, Baltimore, Maryland, USA, July 2003.

[10] G. Peeters. A large set of audio features for sound description (similarity and classification) in the CUIDADO project. Technical report, Ircam, Paris, France, Apr. 2004.

[11] L. Rabiner and B.-H. Juang. *Fundamentals of Speech Recognition*. Prentice-Hall, New Jersey, USA, 1993.

[12] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn. Speaker verification using adapted Gaussian mixture models. *Digital Signal Processing*, 10:19–41, 2000.

[13] V. Sandvold, F. Gouyon, and P. Herrera. Percussion classification in polyphonic audio recordings using localized sound models. In *Proc. of 5th International Conference on Music Information Retrieval*, Barcelona, Spain, Oct. 2004.

[14] P. Somervuo. Experiments with linear and nonlinear feature transformations in HMM based phone recognition. In *Proc. of IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume I, pages 52–55, Hong Kong, 2003.

[15] K. Tanghe, S. Dengroeve, and B. De Baets. An algorithm for detecting and labeling drum events in polyphonic music. In *Proc. of First Annual Music Information Retrieval Evaluation eXchange*, London, UK, Sept. 2005. extended abstract.

[16] K. Yoshii, M. Goto, and H. G. Okuno. Automatic drum sound description for real-world music using template adaptation and matching methods. In *Proc. of 5th International Conference on Music Information Retrieval*, Barcelona, Spain, Oct. 2004.

[17] S. J. Young, N. H. Russell, and J. H. S. Thornton. Token passing: a simple conceptual model for connected speech recognition systems. Technical Report CUED/F-INFENG/TR38, Cambridge University Engineering Department, Cambridge, UK, July 1989.

[1] Koen Tanghe is acknowledged for providing additional information concerning the usage of the applications.