DRUMKIT TRANSCRIPTION VIA CONVOLUTIVE NMF

Henry Lindsay-Smith, *

FXpansion Audio UK
London, UK
henry@fxpansion.com

Skot McDonald

FXpansion Audio UK
London, UK
skot@fxpansion.com

Mark Sandler

Centre for Digital Music
Queen Mary University of London
London, UK
mark.sandler@eecs.qmul.ac.uk

ABSTRACT

Audio to midi software exists for transcribing the output of a multi-mic'ed drumkit. Such software requires that the drummer uses multiple microphones to capture a single stream of audio for each kit piece. This paper explores the first steps towards a system for transcribing a drum score based upon the input of a single mono microphone. Non-negative Matrix Factorisation is a widely researched source separation technique. We describe a system for transcribing drums using this technique presenting an improved gains update method. A good level of accuracy is achieved on on complex loops and there are indications the mis-transcriptions are for perceptually less important parts of the score.

1. INTRODUCTION

Recording a full drum kit requires a large number of microphones and an acoustically treated studio room. A multi-sampled drum workstation (MDW) approximates the sound of a realistic drum kit by using a library of samples for each drum kit piece. These are recorded in a studio with a range of velocities for each kit piece and with multiple microphones. The use of an MDW such as FXpansion's BFD2 [1] allows composers to generate realistic drum parts [1]. For a competent drummer, an electronic drum kit (e-drum kit) can be used to transcribe a score into an MDW. In situations where a drummer does not have an e-drum kit, or prefers the feel of their acoustic kit, there is a use-case for a technique for transcribing drums based upon the input of a single microphone, figure 1

Non-negative Matrix Factorisation (NMF) is a source separation technique which has received much research attention in recent years. Simply described NMF is an unsupervised algorithm which factorises an unknown signal into sources and a set of time varying gains for the sources. Applying NMF to the context of a drummer and recorded drum kit audio we can assign each kit piece to be a source and the drum score (groove) is the set of time varying gains. We will seed the algorithm with sample audio from each kit piece of the drum kit as the sources. The room, microphone and sources will not change and the consistency of this setup should allow us to extract an accurate transcription from the time varying gains.

2. BACKGROUND

2.1. NMF

NMF has been covered in widely in the literature. We refer the reader to [2] for an explanation of NMF and its extension, convolutive NMF (cNMF). To summarise briefly; we transform our test audio into a M x N time frequency matrix $\mathbf{V} \in \mathbb{R}^{\geq 0, M \times N}$. Our goal is then to approximate it using the product of two non-negative matrices $\mathbf{W} \in \mathbb{R}^{\geq 0, M \times R}$ and $\mathbf{H} \in \mathbb{R}^{\geq 0, R \times N}$. R is the number of sources present, the number of drums in our unknown audio. $\mathbf{W}$ is a matrix of spectral bases and $\mathbf{H}$ is a matrix of time varying gains. We define a model $\mathbf{\Lambda} = \mathbf{W} \cdot \mathbf{H}$. The success of the reconstruction is measured using a cost function, such as the KL divergence, between $\mathbf{V}$ and $\mathbf{\Lambda}$. $\mathbf{W}$ and $\mathbf{H}$ are updated using multiplicative updates.

2.2. Convolutive NMF

cNMF, introduced in [2], extends NMF to the convolutive case by using time-frequency sources so $\mathbf{W}$ is extended to a tensor $\mathbf{W_t}$ with each of the R frequency bases having T frames as well. Thus

$$\mathbf{\Lambda} = \sum_{t=0}^{T-1} \mathbf{W_t} \cdot \overset{\mathbf{t}\rightarrow}{\mathbf{H}} \tag{1}$$

where $\overset{t\rightarrow}{(\cdot)}$ is a column shift operator, not defined here for brevity's sake. $\mathbf{W_t}$ and $\mathbf{H}$ are updated as follows. The $\langle\rangle$ operator is defined as the mean for all t, preventing a biased estimate.

$$h_{ijk} \leftarrow \left\langle h_{ijt} \frac{\sum_{i=1}^{M} w_{ij_t} (v_{ik} / \overset{\leftarrow t}{[\Lambda]}_{ik})}{\sum_{i=1}^{M} w_{ij_t}} \right\rangle, \forall t \tag{2}$$

$$w_{ijk} \leftarrow w_{ijt} \frac{\sum_{k=1}^{T} (v_{ik} / [\Lambda]_{ik}) \overset{t\rightarrow}{h}_{jk}}{\sum_{k=1}^{T} \overset{t\rightarrow}{h}_{jk}} \tag{3}$$

* This work was funded by the EPSRC as part of the ImpactQM project EP/H500162/1.

[1] http://www.fxpansion.com/bfd2

Figure 1: *Drum kit mono recording setup. A single overhead microphone captures the signal from all kit pieces*

2.3. Sparsity Constraints

Various researchers have postulated that enforcing a sparsity [3] on the gains improves the factorisation by ensuring more relevant information is captured in $\mathbf{W_t}$ and $\mathbf{H}$ stays sparse and impulse like. One of the side effects of this is the creation of over-complete time-frequency bases, this will prove to be a problem for us. In [4] O'Grady presents an extended algorithm, sparse convolutive NMF (scNMF) by introducing an additional term, $\lambda \sum_{jk} h_{jk}$, to the cost function. This enforces sparsity by minimising the L_1 norm of the elements of the time varying gains. From this new versions of the multiplicative update rules are derived, which we reproduce in 4 and 5. We set have the beta parameter from [4] to 1 and simplified the update rules correspondingly.

$$h_{ijk} \leftarrow \left\langle h_{ijt} \frac{\sum_{i=1}^{M} \bar{w}_{ijt}(v_{ik}/\overset{\leftarrow t}{[\Lambda]}_{ik})}{\sum_{i=1}^{M} \bar{w}_{ijt} + \lambda} \right\rangle, \forall t \tag{4}$$

$$w_{ijt} \leftarrow w_{ijt} \frac{\sum_{k=1}^{T} \overset{t\rightarrow}{h}_{jk}[(v_{ik}/\Lambda_{ik}) + \bar{w}_{ijt}(\bar{w}_{ijt})]}{\sum_{k=1}^{T} \overset{t\rightarrow}{h}_{jk}[\bar{w}_{ijt}(\bar{w}_{ijt}(v_{ik}/\Lambda_{ik}))]} \tag{5}$$

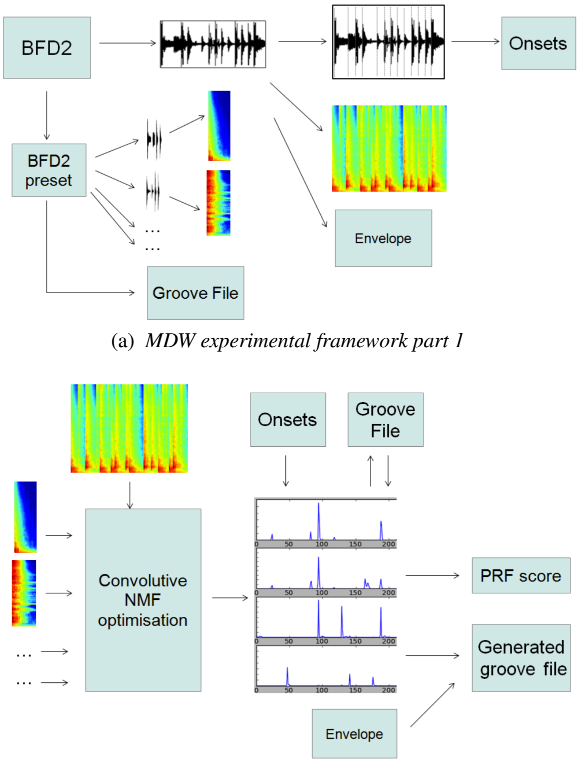
$\bar{\mathbf{W}}_\mathbf{j} = \frac{\mathbf{w_j}}{||\mathbf{w_j}||}$ is the normalised version of $\mathbf{W_j}$, calculated after each update.

2.4. Percussion transcription using NMF

Previous work by Paulus [5] used NMF to factorise simple drum loops with kick, snare and closed hi-hat and a wide onset tolerance of 30ms either side of the onset. Good results of 96% hit accuracy were reported. Other work in the field has tended to focus percussion transcription systems upon extracting drums from polyphonic music [6], [7], [8].

2.5. Drummer transcription

As described in section 1, and illustrated in figure 1, a mono mic transcription system would be of use to home studio musicians with rudimentary equipment. The drummer can easily sample audio for each kit piece and articulation they wish to use as sources and also perform the drum tracks they wish to transcribe. Articulations are defined as different sounds by the same kit piece, for example open and closed hi-hat or snare rim shot and center hit. The room, microphone and microhone positioning are constant and by working with a constrained setup in this way we hope to be able to achieve the high levels of accuracy we desire.



(a) *MDW experimental framework part 1*

(b) *MDW experimental framework part 2*

Figure 2: *MDW experimental framework*

3. TRANSCRIPTION OF A MDW GENERATED DRUM LOOP

MDWs allow us to generate realistic drum parts and simulate a drummer with a single microphone. Because the groove is programmed within the software we have access to the ground truth and the individual audio samples used to generate it. This removes the need for annotation of the drum loops. This paper extends the work done by Paulus [5]. For a percussion transcription system to be useful, we are adding three further requirements; the onset accuracy must be 10x more accurate (3ms), the system should be able to deal with simple articulations and we should attempt to recover the velocities of the hits.

In addition it is desired that the system be able to run in near real-time. cNMF and scNMF and our own optimisation, cdNMF, were evaluated. Convolutive NMF variants are good candidate algorithms because percussion events should be well modeled by fixed length time-frequency bases. This contrasts with pitched insturments which have varying note lengths.

4. EXPERIMENTAL FRAMEWORK

4.1. The test drum loop

BFD2 was used to render a drum loop from a groove using just the left overhead mic channel. The rendered audio, which was at a sample rate of 44.1kHz, was transformed into a time-frequency representation **V** as described in 2.2 and 2.3. The short time fourier transform (STFT) and mel frequency representations were used, in both cases based upon a window length of 2048 and with a 75% overlap. To compress the STFT to a mel frequency representation a range of equally spaced, overlapping, triangular critical band filters were created on the mel scale defined by 6.

$$Z_{mel}(f_{kHz}) = log(1 + f/0.7) * 1127.01048 \tag{6}$$

A modified spectral difference onset detector, based upon [9], was used to extract all the onset times from the rendered loop. The modification was made in order to achieve an improved accuracy in the onset positioning. The explanation of the modification is beyond the scope of this paper. The amplitude envelope was also extracted from the loop using a simple envelope detector [10]. The BFD2 preset also provided the kit piece (source) audio files and the groove file from which we extracted the ground truth onset times and classifications. The grooves contained hits at multiple velocities and consequently BFD2 uses multiple source files of different amplitudes when rendering the drum loop.

4.2. The decomposition sources

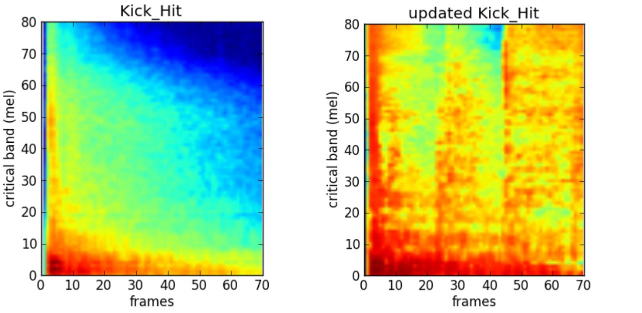
For each source we chose four files extracted from the BFD2 sample library from .7 to 1.0 of the maximum amplitude. The files were selected from the same kit that was used to render the drum loop. The source audio files were subject to the same time-frequency transformation as the rendered loop and the mean of the transformations of the different amplitudes was used. We found this yielded better results than using a single file per source. The combination of the mean of the transformations for all sources was $\mathbf{W_t}$. This part of the framework is illustrated in figure 2a.

4.3. The optimisation

In order to seed the optimisation with an initial low cost **H** was initialized with 1 at every expected onset point and 0.1 at all other points. **H** was normalised for each of the R sources after the multiplicative updates were finished. Rather than try to detect the onset positions from these gains, we simply attempted to detect if there was a valid onset by the use of an empirically determined threshold. The threshold was kit piece and time-frequency transform specific. The onset position was then mapped to the closest onset previously detected. With our onset times and classes established we calculated a precision, recall and F-measure score for the transcription. A reconstructed groove file was generated from the gains with the dynamics provided by taking the mean of the gain strength at each onset and the previously extracted amplitude envelope. This groove file enabled us to play the transcription in BFD2. This part of the framework is illustrated in figure 2b.

Our experiments uncovered an improvement which lead to a very impulse like gain structure by modifying equation 2 to 7.

$$h_{ijk} \leftarrow \left\langle h_{ijt} \frac{\sum_{i=1}^{M} w_{ij\,t}(v_{ik}/[\overset{\leftarrow t}{\Lambda}]_{ik})}{\sum_{i=1}^{M}\sum_{i=1}^{T} w_{ij\,t}} \right\rangle, \forall \tag{7}$$



(a) *Mel spectrogram of kick drum* (b) *Same kick drum updated by cNMF, showing existence of more than one hit*

Figure 3: *Kick drum before and after updating demonstrating the change to an overcomplete basis*

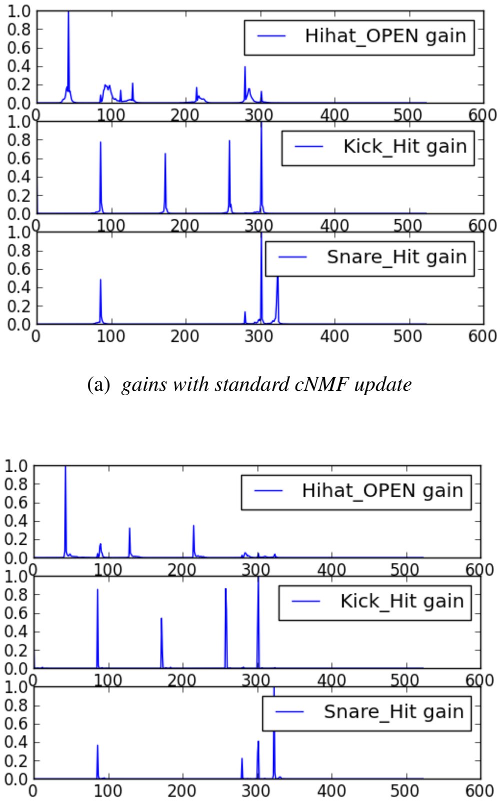
The simple conversion of the denominator into a constant value across the t's weights the gain update towards the frames of $\mathbf{W_t}$ which have the largest magnitude. We call this update constant denominator NMF (cdNMF). An example of the improvement is shown in figure 4.

5. RESULTS

Three sets of data were used; dataset A - simple loops with only kick, snare and hi-hat, dataset B - more complex loops with kick, snare, hi-hat, tom-toms and a cymbal, dataset C - complex loops with kick, snare, hi-hat, tom-toms and 2 articulations on the snare and hi-hat. Each dataset consisted of 5 loops. The datasets were evaluated against cNMF, scNMF and cdNMF. An STFT and mel frequency spectrums with 80 and 160 critical bands were used. For both cNMF, scNMF and cdNMF the algorithms were run with the updates to $\mathbf{W_t}$ turned on and off. It was found that the updates to $\mathbf{W_t}$ did not help the transcription. The ground truth solution contains overlapping basis functions, as the tail of one drum hit sustains into the onset of the next. Unfortunately this is not optimal solution when minimising our cost function. As highlighted in figures 3a & 3b, the updates to $\mathbf{W_t}$ move towards an over-complete basis with the optimisation trying to capture more than one hit per source. We used a wide onset threshold of 30ms for the cdNMF method. Experiments using a more musically useful window of 3ms were run against all methods. The results are presented in table 1 with all tests running for 30 iterations.

6. DISCUSSIONS

Our scenario would benefit from a larger dataset and a more detailed exploration of all the available parameters, however we can make some preliminary observations. Decreasing the onset detection window from 30ms to 3ms had a detrimental effect on our F measure. Unfortunately, to produce a drum score which captures the rhythm and feel of a drummer we need this level of accuracy. Increasing the complexity of the datasets results in a significant drop in accuracy, however in informal listening of the recovered grooves the similarity seemed higher than the F-measure results suggested. This can be attributed to the loss of accuracy taking place where there are simultaneous hits or low velocity hits, which



(a) *gains with standard cNMF update*

(b) *gains with cdNMF update*

Figure 4: *Gains with cNMF and cdNMF demonstrating improved sparsity for cdNMF*

are perceptually less important. No formal evaluation of the dynamics were conducted but an informal evaluation by the authors revealed that the rendered grooves with dynamics were preferable to grooves without dynamics. In general the division of the source data into datasets A, B and C was arbitrary with the number of closely positioned onsets and number of simultaneous hits having a large effect on the transcription accuracy. Our feelings are that the algorithm works well enough on all but the most complicated loops and that a drummer could learn the limits of the system.

The STFT was the most successful transform but at a considerable time penalty. The mel 80 transform offered good transcription accuracy and approximately 12 times faster performance, taking roughly 15 seconds to set up the scenario and a further 10 to optimise on a desktop workstation in our python implementation. An alternative time-frequency transform, tailored for drums, might improve the transcription.

Table 1: *F measure score for the different optimisations and datasets and onset window size and time-frequency transforms*

STFT	*cNMF*	*scNMF*	*cdNMF*
Dataset A 30ms	0.70	0.87	0.97
Dataset A 3ms	0.55	0.53	0.79
Dataset B 30ms	x	x	0.85
Dataset B 3ms	0.41	0.40	0.59
Dataset C 30ms	x	x	0.82
Dataset C 3ms	0.33	0.28	0.50
mel 160	*cNMF*	*scNMF*	*cdNMF*
Dataset A 30ms	0.80	0.81	0.96
Dataset A 3ms	0.60	0.60	0.78
Dataset B 30ms	x	x	0.86
Dataset B 3ms	0.51	0.42	0.63
Dataset C 30ms	x	x	0.80
Dataset C 3ms	0.34	0.36	0.48
mel 80	*cNMF*	*scNMF*	*cdNMF*
Dataset A 30ms	0.7	0.79	0.96
Dataset A 3ms	0.56	0.61	0.76
Dataset B 30ms	x	x	0.86
Dataset B 3ms	0.42	0.49	0.57
Dataset C 30ms	x	x	0.81
Dataset C 3ms	0.31	0.33	0.5

7. REFERENCES

[1] maragos g, “Long distance recording using virtual drums,” M.S. thesis, VFCC Pennsylvania, 2011.

[2] P. Smaragdis, “Non-negative matrix factor deconvolution; extraction of multiple sound sources from monophonic inputs,” *Independent Component Analysis and Blind Signal Separation*, pp. 494–499, 2004.

[3] J. Eggert and E. Korner, “Sparse coding and nmf,” in *Neural Networks, 2004. Proceedings. 2004 IEEE International Joint Conference on*. Ieee, 2004, vol. 4, pp. 2529–2533.

[4] P.D. O’Grady and B.A. Pearlmutter, “Discovering speech phones using convolutive non-negative matrix factorisation with a sparseness constraint,” *Neurocomputing*, vol. 72, no. 1-3, pp. 88–101, 2008.

[5] J. Paulus and T. Virtanen, “Drum transcription with non-negative spectrogram factorisation,” in *Proceedings of the 13th European Signal Processing Conference*, 2005, p. 4.

[6] D. Fitzgerald, M. Cranitch, and E. Coyle, “Using tensor factorisation models to separate drums from polyphonic music,” 2009.

[7] K. Yoshii, M. Goto, and H.G. Okuno, “Drum sound recognition for polyphonic audio signals by adaptation and matching of spectrogram templates with harmonic structure suppression,” *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 15, no. 1, pp. 333–345, 2007.

[8] O. Gillet and G. Richard, “Transcription and separation of drum signals from polyphonic music,” *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 16, no. 3, pp. 529–540, 2008.

[9] J.P. Bello, L. Daudet, S. Abdallah, C. Duxbury, M. Davies, and M.B. Sandler, “A tutorial on onset detection in music signals,” *Speech and Audio Processing, IEEE Transactions on*, vol. 13, no. 5, pp. 1035–1047, 2005.

[10] Udo Zolzer, *DAFX, digital audio effects*, Wiley, 2nd, edition, 2011.