Rhythm transfer in music

Somesh Ganesh and Alexander Lerch

Abstract—**This paper is a half-yearly update for a one year master's project. The concept of automatic rhythm transfer in music using non-negative matrix factorization is explained and proposed. In this approach, NMF will be used to extract the activations of the rhythmic components in an audio track which will then be fed to a system where this activation matrix will be manipulated to be rhythmically similar to a target rhythm. The current algorithm, results, limitations and future work have been discussed in this paper.**

Index Terms—**rhythm, non-negative matrix factorization, music, transfer**

I. Introduction

The concept of non-negative matrix factorization (NMF) has been applied extensively for many music information retrieval (MIR) tasks. In this paper, we propose the use of non-negative matrix factorization for the task of rhythm transfer in music. The idea is to have an input audio track and automatically modify it in a way to mimic or resemble the rhythm of a target audio track. Sections II & III discuss the motivation for this project and previous work around this topic and NMF. Sections IV & V go over the algorithm and evaluation for the proposed system. Section VI goes over the results and discussion so far. Sections VII & VIII discuss the novelty in this project and the deliverables at the end of the project. Section IX will be the timeline for the remainder of the project and this paper will conclude in Section X.

II. Motivation

The concept of style transfer in audio is an exciting domain to explore. This is because of the prospect of using MIR in the creative domain [1]. Also, the concept of breaking down signals into their basis functions using the concept of NMF has motivated the author to explore some sort of style transfer using this technique. This concept, if implemented successfully, can have potential in academic research (in fields such as music information retrieval and generative models for audio) and commercial applications as well (this would include music creation, performance and listening experience).

III. Related work

To the knowledge of the author, the proposed task in this paper has not been attempted yet. This section covers some work in topics like drum transcription, automatic mash-up of songs and audio source separation using NMF. All of these provide inspiration for our task. This will become much clearer in section IV.

Yoshi et al. [2] use a template adaptation and matching technique to transcribe the bass drum and snare drum in an audio track. The algorithm uses spectrograms of the input audio as templates for transcription. Yoshi et al. [3] use this same transcription technique to develop a drum sound equalizer for controlling the volume and timbre of the bass drum and snare drum in the track. Using these learned templates, the user of the application can individually increase/decrease each drum hit's volume and/or replace it with another drum sample (this would involve replacing the template of the sample to be replaced with the template of the new sample). This application goes one step further into changing rhythms by allowing the user to change the location and volume of each drum hit in each bar manually using a GUI (graphical user interface). Yoshi et al. [4] further this approach by transcribing the bass drum, snare drum and hi-hat samples in an audio track. This approach uses Goto's distance [5] as a similarity measure and uses some harmonic structure suppression techniques. However, in all these techniques, the user manually changes some parameters in the input audio. There is no automatic change to achieve a certain "target" sound.

Davies et al. [6] introduced a system, named "AutoMashUpper", which would automatically create song mash-ups. The idea is to have an input song segmented on the phrase level and have its similarity measured with a list of candidate songs. The similarity measure takes into account harmonic compatibility, rhythmic compatibility and spectral balance. The candidates are then be ranked according to this measure and the ones with the highest probability are chosen for the mash-up.

Smaragdis [7] explored and introduced the concept of using NMF for polyphonic music transcription. Spectrograms of the input audio are fed as input to the transcription system. The system returns the different templates present in the input audio along with its activations throughout the whole track. Then onward, NMF was used in a lot of music transcription tasks. [8] introduced a variant of NMF called NMFD (non-negative matrix factor deconvolution) which also accounts for temporal information in the template matrix. Some recent drum transcription systems using NMF include [9] & [10]. These papers implement PFNMF (partially fixed NMF) along with a template adaption technique, thus allowing to separate out certain desired components (in this case, the bass drum, snare drum and hi-hats) which allows for a good drum transcription accuracy with a minimal training dataset. Dittmar et al. [11] implement a score-informed separation and restoration of drum recordings using NMFD. The score-informed separation technique imposes some constraints on the algorithm used.

This paper is a progress report for the Master's project of Somesh Ganesh for fall 2017.

Somesh Ganesh is a masters student in the music technology program at the Georgia Institute of Technology, Atlanta, GA 30318 USA (email: someshg94@gatech.edu)

Alexander Lerch is Somesh's advisor for this project and the head of the music informatics laboratory at the Georgia Tech Center for Music Technology, Atlanta, GA 30318 USA (email: alexander.lerch@gatech.edu)

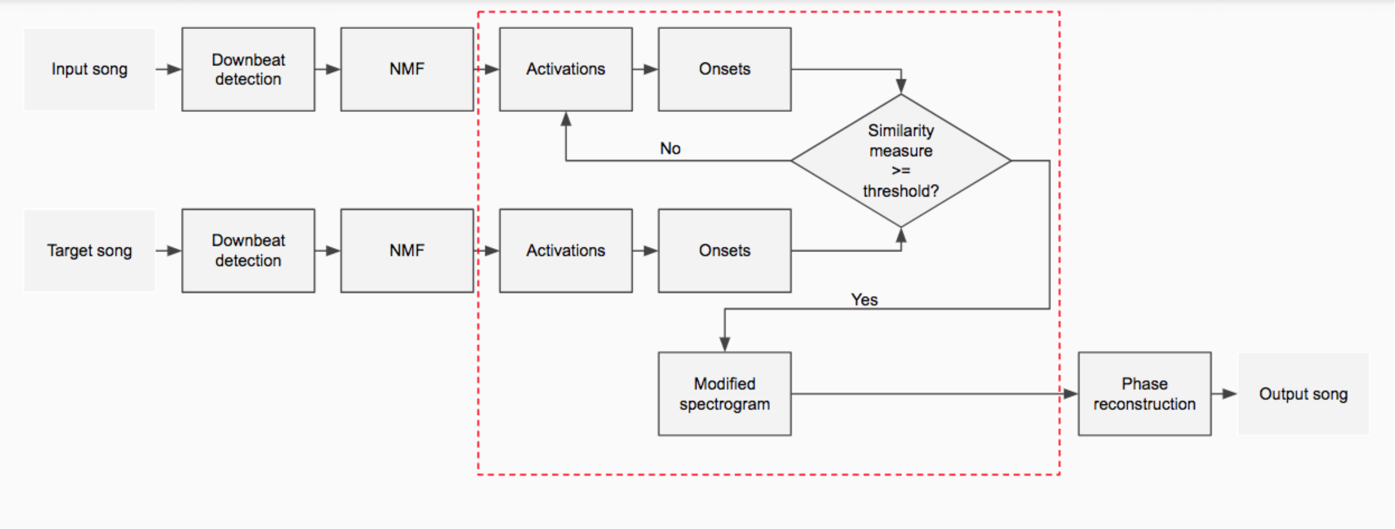

Fig. 1. Proposed algorithm

Laroche et al. [12] propose a drum extraction system using multi-layer NMFD. This recent publication uses NMFD and hence, stores temporal information of the different templates detected and transcribed. Driedger et al. [13] propose a system to reconstruct an audio track using NMF and playing around with the template matrix. This paper is the main inspiration to this proposed rhythm transfer project. Instead of working on the templates, this project proposes to work on the activation matrix. Hence, while maintaining the sound of the templates, the rhythm of a track is changed by operating on the activation matrix.

IV. Proposed Algorithm

A block diagram of the proposed algorithm has been shown in figure 1.

A. *Downbeat detection*

The input and target audio files are first passed through a downbeat detection block where the downbeat locations are used to segment the audio files on the bar-level.

B. *NMF*

Each bar in the audio files is then passed through a drum transcription algorithm using NMF. Here, the activation matrix for templates that contribute to the rhythmic content in the files is extracted. Since this process is done for the input song as well as the target song, there are two activation matrices.

C. *Activations*

Here, the input activations are warped to the target activations and are modified in a way so as to pass a rhythm similarity measure test with the target activations. This automatic processing can be done using copying and pasting blocks in the input matrix to the desired locations. This method is explained in more detail in section V-A.

D. *Onsets*

These activations are now passed to an onset detection block. The input (bar) is quantized to a discrete number bin representation and onsets are detected per bin.

E. *Similarity measure*

A similarity measure is computed between the two onset functions (input and target). If the measure does not pass a threshold test, the input activation matrix is modified again till it passes this test. If it passes the test, it exits this loop to proceed to the next block.

The threshold value for the similarity test here dictates to what extent the input activation is processed. If the threshold is at a value of 0, the input activations are not modified at all. As the value of the threshold increases all the way to the maximum, the input activations are meant to resemble the target activations to a greater extent.

The distance measure used to measure similarity here is a modified swap distance. Swap distance is defined as the number of swaps needed to convert one binary string to another. Since this distance works only on strings of the same length, we modify this distance to accommodate strings of different lengths (since number of onsets in the input and target could be different). This modification involves making sure that the every point in the longer string is mapped to at least one point in the shorter string, and that every point in the short string has at least one point in the long string mapped to it.

F. *Modified spectrogram*

After the input activations pass the similarity test, a new spectrogram is constructed using templates of the input audio file and the modified input activations.

G. *Phase reconstruction*

The final block in the algorithm is phase reconstruction. Reconstruction of the modified audio file requires the new complex spectrogram. For this, the phase of the input or target file would not be ideal. The phase for the new magnitude spectrogram is computed and used to reconstruct the audio file in the time domain.

V. Proposed evaluation

A. Methodology

The blocks inside the dotted rectangle from figure 1 are implemented from scratch. The whole process till now has been implemented in MATLAB. Details of the current implementation are given below.

- Data: The audio files used are all of bar-length and created by the author. This decision was made to avoid any ambiguity caused in the process by the downbeat tracker.
- NMF: The toolbox from [**?**] has been used to perform drum transcription[1]. The bass drum, snare drum and hi-hats are transcribed and their activations are obtained.
- Onset detection: Onsets from the activations for these three drum sounds are detected. Each activation row (i.e., activation for one sound) is quantized to 32 bins, thus producing a 32x3 = 96 dimension feature vector.
- Similarity measure: As stated in section IV-E, the modified swap distance is used to measure similarity between the input and target rhythmic patterns. There is currently no threshold parameter set. The input activations are meant to completely mimic the target rhythm.
- Activation processing: Currently, the input activations are processed by simply copying and pasting blocks in different locations. All input locations are mapped to the closest target onset. The original location of the input onset is then replaced with zeros.
- Phase reconstruction: Currently, no phase reconstruction has been implemented. The phase of the input signal is being used to reconstruct the modified audio file.

B. Metrics

The system is currently being evaluated by comparing plots of the input, target and modified audio files in the time domain and their activations. Subjective evaluation is being done by listening to the output of the system. The results are elaborated in the upcoming section.

There a couple of important questions that need to be explored here which would involve subjective listening tests as the main evaluation measure.

- **How rhythmically similar are the output and target songs?** The aim of the project is to modify the input audio in such a way that its rhythm resembles the rhythm of the target audio.
- **Is there something unique to the output rhythm?** We don't want the output song to completely mimic the rhythm of the target song. It should still retain some of its characteristics after the whole process. For example, simply switching the activation matrices from the target audio is not the desired result here.

[1] https://github.com/cwu307/NmfDrumToolbox

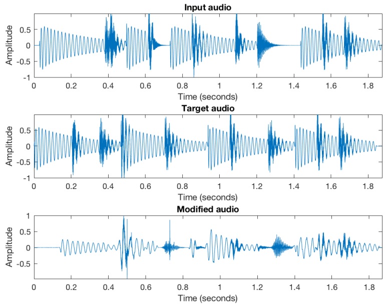

Fig. 2. Audio files in time domain

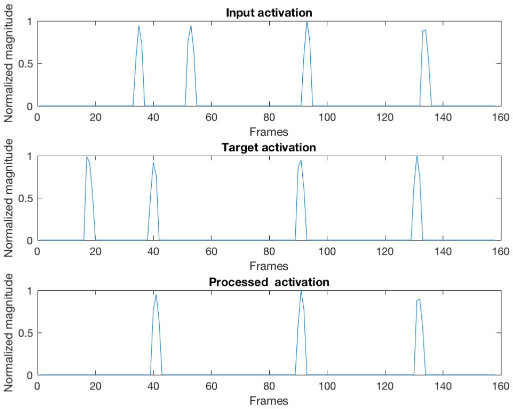

Fig. 3. Hi-hat activations (1 bar)

VI. Results and discussion

Let us look at some of the results for this implementation. We will take one input audio file and one target audio file and go through the activation processing and do some analysis.

Figure 2 shows the input, target and modified audio files, all in the time domain. The drum sounds in these files are reasonably visible from the time domain plots. Each audio file, the input and the target contain 4 onsets for each drum sound (bass drum, snare drum and hi-hat). From these plots, we can see that the modified audio has transients at locations corresponding to the locations of transients in the target file.

The activations for the three drum sounds for one bar are shown in figures 3, 4 and 5.

From all these activation plots, it is visible that the activations from the input are being mapped to their closest activation locations from the target. All the activation plots have been thresholded above a certain value to make the onsets clearly visible. It is also visible that the onset detection is not extremely robust (some onsets in the first bin of the bar are not detected). Input onset locations which do not resemble the target onset locations are replaced with zeros. This may not be ideal and a smoothing function for this must be implemented.

While listening to most of the outputs and from looking at the plots, it can be clearly made out that the phase is a pressing issue here. Phase reconstruction is a high priority task for this project since the output must sound pleasing to the user.

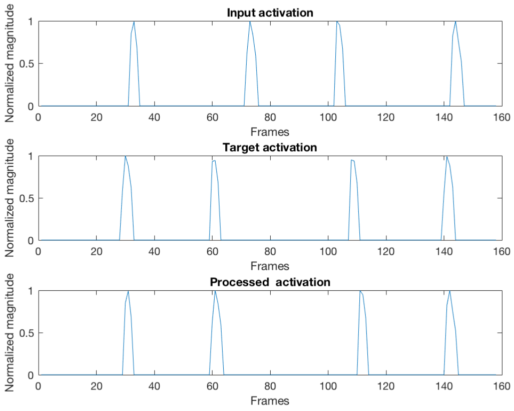

Fig. 4. Snare drum activations (1 bar)

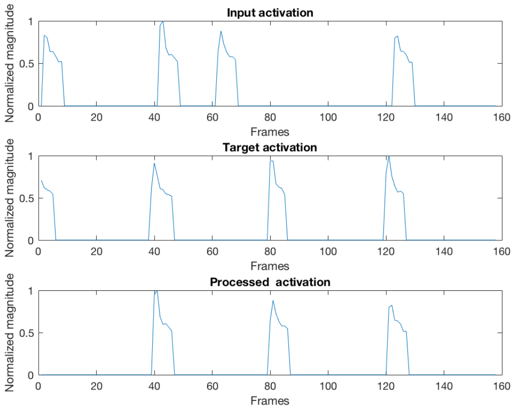

Fig. 5. Bass drum activations (1 bar)

Although there are issues with the current implementation, while listening to the outputs, one can make out that the modified audio file has a rhythm similar to the target file. This provides reason to believe that this concept of rhythm transfer using NMF can produce promising results.

The limitations of this project so far have been discussed. Taking all of these into consideration, the following future work has been listed in decreasing order of priority.

- Phase reconstruction is of the highest priority and is must be implemented to help in the evaluation process.
- The smoothing of the activations in locations where zeros are added must be implemented.
- The threshold parameter must be integrated. The threshold parameter dictates the amount of modification on the input activations. It would be interesting to listen to the output for varying values of this threshold parameter.
- So far, testing has been done on audio files containing only these drum sounds. Audio files with more instruments must also be used for testing. This may involve tweaking parameters in the NMF algorithm to achieve better performance.
- The downbeat tracker must be integrated into the system so that testing can start on real world signals rather than simple loops.
- An interactive user interface would allow for better evaluation.
- Different distance measures could be implemented and the activations of more rhythmic components can be processed.

VII. Novelty

The novelty in this project lies in the automatic manipulation of an audio file to make it rhythmically similar to a target file. As mentioned before, this sort of system has not been developed before. Hence, this project could pave the way for future research in rhythm learning and transfer in music. Till now, softwares like [3] allow the user to visualize the rhythm and manually make changes to it. The idea of a system being able to understand different rhythms and automatically change the rhythm of a song would be a step forward in the creative music information retrieval research community. This could also aid in other research areas like algorithmic composition and generative models for audio.

VIII. Deliverables

All the code for the current implementation can be found online[2]. At the end of the project, an open source toolbox will be made available online for anyone to use. Users will be able to input audio and listen to the changed rhythm of the audio depending on a target rhythm. An interactive user interface would be ideal for such a use case. A paper detailing the algorithm, implementation, experiments and evaluation will also be made available.

IX. Proposed timeline

The proposed timeline for the remainder of this project is as follows

- **January**: Implement phase reconstruction for modified spectrogram, smoothing the activations at locations where currently zeros are being added.
- **February**: Integrate threshold parameter for the similarity measure test and activation processing, integrate downbeat tracker and start developing user interface.
- **March**: Complete user interface and begin listening tests (if time permits, consider adding more rhythmic components for activation processing).
- **April**: Complete listening tests, analysis and report.

X. Conclusion

In this paper, the concept of rhythm transfer in music using NMF has been introduced and the approach to build the system and evaluate it has been laid out. The idea is to use NMF to extract activations of the rhythmic components in an audio file and modify these activations to resemble a target rhythm. Listening tests would be the main evaluation measure for this project. The progress so far shows that further experiments and improvements could lead to more promising and successful results.

[2] https://github.com/someshganesh94/RhythmTransfer

Acknowledgment

I would like to thank Alexander Lerch, Mastaka Goto, Chih-Wei Wu and Christian Dittmar for their guidance and inspiration during this project.

References

[1] X. Serra, M. Magas, E. Benetos, M. Chudy, S. Dixon, A. Flexer, E. Gómez, F. Gouyon, P. Herrera, S. Jordà *et al.*, "Roadmap for music information research. creative commons by-nc-nd 3.0 license isbn: 978-2-9540351-1-6," 2013.

[2] K. Yoshii, M. Goto, and H. G. Okuno, "AdaMast: A Drum Sound Recognizer based on Adaptation and Matching of Spectrogram Templates."

[3] K. Yoshii, M. Goto, and H. Okuno, "INTER:D: a drum sound equalizer for controlling volume and timbre of drums," in *2nd European Workshop on the Integration of Knowledge, Semantics and Digital Media Technology (EWIMT 2005).* IET, 2005, pp. 205–212.

[4] 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," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 15, no. 1, pp. 333–345, jan 2007.

[5] M. Goto and Y. Muraoka, "A sound source separation system for percussion instruments," *Transactions of the Institute of Electronics, Information*, 1994.

[6] M. E. Davies, P. Hamel, K. Yoshii, and M. Goto, "Automashupper: Automatic creation of multi-song music mashups," *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, vol. 22, no. 12, pp. 1726–1737, 2014.

[7] P. Smaragdis and J. Brown, "Non-negative matrix factorization for polyphonic music transcription," in *2003 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (IEEE Cat. No.03TH8684).* IEEE, pp. 177–180.

[8] P. Smaragdis, "Non-negative Matrix Factor Deconvolution; Extraction of Multiple Sound Sources from Monophonic Inputs International Congress on Independent Component Analysis and Blind Signal Separation (ICA) Non-negative Matrix Factor Deconvolution; Extraction of Multip," 2004.

[9] C.-W. Wu and A. Lerch, "Drum transcription using partially fixed non-negative matrix factorization with template adaptation," in *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR).* Malaga: ISMIR, 2015.

[10] ——, "Drum transcription using partially fixed non-negative matrix factorization," in *Proceedings of the European Signal Processing Conference (EUSIPCO).* Nice: EURASIP, 2015.

[11] C. Dittmar and M. Muller, "Reverse engineering the amen break: score-informed separation and restoration applied to drum recordings," vol. 24, no. 9, pp. 1531–1543, 2016.

[12] C. Laroche, H. Papadopoulos, M. Kowalski, and G. Richard, "Drum extraction in single channel audio signals using multi-layer Non negative Matrix Factor Deconvolution," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).* IEEE, mar 2017, pp. 46–50.

[13] J. Driedger, T. Prätzlich, and M. Müller, "Let It Bee – towards NMF-inspired audio mosaicing," in *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, Malaga, Spain, 2015, pp. 350–356.