TIME SCALE MODIFICATION OF AUDIO USING NON-NEGATIVE MATRIX FACTORIZATION

Gerard Roma
CeReNeM
University of Huddersfield
Huddersfield, UK
g.roma@hud.ac.uk

Owen Green
CeReNeM
University of Huddersfield
Huddersfield, UK
o.green@hud.ac.uk

Pierre Alexandre Tremblay
CeReNeM
University of Huddersfield
Huddersfield, UK
p.a.tremblay@hud.ac.uk

ABSTRACT

This paper introduces an algorithm for time-scale modification of audio signals based on using non-negative matrix factorization. The activation signals attributed to the detected components are used for identifying sound events. The segmentation of these events is used for detecting and preserving transients. In addition, the algorithm introduces the possibility of preserving the envelopes of overlapping sound events while globally modifying the duration of an audio clip.

1. INTRODUCTION

Time-scale modification (TSM) of audio signals is nowadays an essential audio processing tool in music and audio production. TSM became particularly popular in music creation workflows based on the reuse of readily available audio. A common goal in this context is to stretch audio clips, that often contain their own rhythmic micro-structures, so that they will match a given musical context. It can be argued that the key aspect for preserving the structure is the location of sound events in time. Existing TSM algorithms will also change the duration of the sounds themselves, for instance, the duration of a drum hit, which is not necessarily desirable and may sound unnatural. On the other hand, artifacts of TSM algorithms are used as musical features in electronic music experimentation. Different algorithms will produce different kinds of artifacts so they can be creatively abused to produce different sound effects. In the end, a common situation in music production software is to choose between different TSM algorithms that may perform differently for different material.

One difficulty in TSM is the presence of overlapping sound events of different durations and temporal structures. Given recent advances in audio source separation research, we are interested in whether source separation algorithms could help with TSM. Even when separation is not perfect, mixing the estimates of sound sources back together often helps to diminish any artifacts introduced in the separation. This feature can be used to improve TSM by allowing separate scaling of the component sounds of an audio excerpt.

A key algorithm often used for source separation is non-negative matrix factorization (NMF). NMF is an unsupervised method, which has been shown to produce good results for transcription and separation of signals with a clear percussive profile, such as piano sounds [1] or drums [2]. Since it has to learn from the signal (typically the magnitude spectrogram), its effectiveness depends largely on the invariants present in the structure of the data, as the algorithm tries to reconstruct it from a limited set of spectral patterns and their activations. In this sense, while NMF is limited for dealing with polyphonic music, it can often cope well with short sounds such as the musical loops used for some types of rhythmic music. NMF will essentially capture repetitions of spectral patterns, so, for example, in drum patterns it will capture the structure of activations of drum sounds. Similarly, in the case of pitched sounds with fixed spectrum such as piano notes, it can easily capture the structure of tonal melodies.

In this paper we investigate the use of NMF for TSM. By separating different sound events, we expect NMF to make it possible to stretch them separately, allowing the envelope of overlapping events, such as percussive instruments and resonant bodies, to be perceived more naturally. In addition, by introducing a new way of modifying the duration of audio signals, our algorithm introduces a different kind of artifact for materials on which the algorithm fails to produce natural sounding results, which in turn may yield new affordances for sound design and music. Specifically, when NMF fails to identify note-like events, the proposed algorithm may produce rhythmic modifications or even more extreme misplacing of parts of the signal. These artifacts could no doubt be creatively abused by experimenters seeking new sonorities.

The paper is organized as follows. In the next section, we briefly review related work in the field of TSM. In Section 3, we describe our proposed algorithm for TSM using NMF. In Section 4 we discuss some examples that illustrate the potential of the algorithm. Finally, we draw some conclusions and discuss future work.

2. RELATED WORK

Research in TSM was relatively active between the 1980s and early 2000s. Algorithms developed back then, such as time domain overlap and add (OLA), waveform similarity overlap and add (WSOLA) [3], or the phase vocoder [4], remain popular. The latter two algorithms provide relatively good results for music content, but introduce very strong artifacts when stretching transients. Both WSOLA and the phase vocoder have been thus improved with transient detection [5, 6, 7]. Academic research on TSM considerably slowed down afterwards. Perhaps due to the success of TSM for music and audio production in digital audio workstations, industry took the lead. During the last few years, research on TSM has been growing again. This may be partially due to the potential of audio source separation research. To some extent, growing interest in reproducible research in audio signal processing can also be credited for the renewed interest, since algorithms used in com-

mercial products are often not well known outside the companies that make them.

A significant recent contribution was provided in [8] by applying harmonic-percussive source separation (HPSS) and then using OLA for percussive estimates and the phase vocoder for harmonic estimates. The authors also presented the Matlab TSM toolbox, including the classic algorithms and their own, in [9]. The algorithm proposed in [10] applies a similar concept, but unlike [8] it is able to keep transient processing in the frequency domain.

In this paper, we explore the use of another source separation technique, NMF, which allows different treatment of different sources more generally than transient/harmonic separation. The activation curves resulting from the NMF separation are used as a cue for the presence of transients due to specific components. Our approach is inspired by the system presented in [5]. An important detail is that, unlike in the classic phase vocoder, the system in [5] proposed the use of the same hop size for both the analysis and synthesis stages. Using the same hop size allows using the (slower) NMF decomposition as part of the analysis stage, while different scaling factors can be tried in the synthesis stage without the need to analyze again. It also allows tuning all the windowing parameters as suitable for the input material.

3. PROPOSED ALGORITHM

3.1. Overview

We now briefly describe the proposed system. A block diagram is shown in Figure 1. The system is intended for time-scale modification of relatively short (e.g. a few seconds) audio signals. The time domain audio signal is first converted to a spectrogram via the short-time Fourier transform (STFT). The magnitude spectrogram is then decomposed into several components via NMF. As per the NMF framework (described in Section 3.3), each component consists of a basis function and an activation function. Components are then segmented into sound events by analysis of the activation function. The activation function is also used to identify one or more transients within a given sound event. For each of the resulting events, a number of frames are copied verbatim into the synthesized spectrogram. These can be either the detected transients or the whole event, which can be preferred for percussive sounds. For the remaining frames, a new scaling factor is computed in order to respect the scaled duration for the whole event. Time scaling is then applied following the principles of the phase vocoder. The resulting component spectrograms are then mixed and synthesized via inverse STFT.

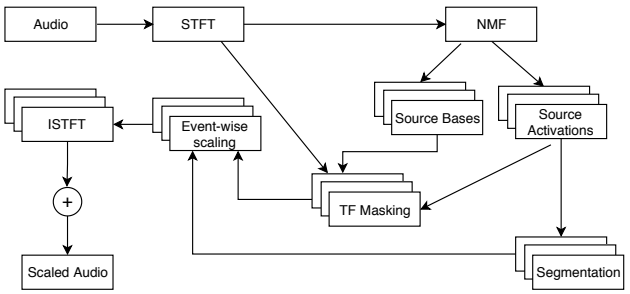

Figure 1: *Block diagram of the proposed system.*

3.2. Source separation

Our system is based on the assumption that the input signal is a mixture of I component signals,

$$x(t) = \sum_{i=1}^{I} x_i(t) \tag{1}$$

Here, the component signal x_i is assumed to have the appropriate gain with respect to the mixture. We want to produce a time-scaled version of x, y, which is analogously a mixture of signals $\sum_i y_i(t)$. Assuming the mixing model holds in the (complex) frequency domain, we estimate the component frequency-domain signals $X_i(k,n)$ from the STFT $X(k,n)$ of x (where k and n are respectively frequency and time indices) via NMF. We then stretch each component X_i into Y_i, and obtain y_i via inverse STFT. The stretched components are then mixed in the time domain.

3.3. NMF decomposition

NMF is typically applied to a magnitude spectrogram. TSM will require using both the magnitude spectrogram, denoted as

$$V = |X| \tag{2}$$

and the phase spectrogram, denoted as

$$\Phi = \angle(X) \tag{3}$$

Under the NMF framework, an approximation of V is obtained as

$$\hat{V} = WH \tag{4}$$

The matrix $W \in \mathbb{R}^{K \times I}$ contains a set of I bases, which typically represent static spectra corresponding to each of the detected sources. The matrix $H \in \mathbb{R}^{I \times N}$ contains a corresponding set of I activations, which represent the temporal envelopes of each component. We can use these activations to find the positions of transients (typically corresponding to note onsets) and general active regions corresponding to each component, thus applying different stretch factors to preserve the structure of sound events. In addition, it is often possible to classify the bases into tonal and percussive sounds [11, 12], which could be used for applying different stretching strategies similar to [8]. However in this paper we just consider the option of preserving the duration of the active region as a user parameter.

For this to work it is of course crucial to obtain a good decomposition that represents the perceived components of the signal. A common strategy is to minimize the divergence

$$D_{KL}(V|WH) = \sum_{kn} d_{KL}(V(k,n)|\sum_{i} W_i(k)H_i(n)) \tag{5}$$

where

$$d_{KL}(x|y) = x \log \frac{x}{y} - x + y. \tag{6}$$

As originally proposed in [13], this can be done via a simple multiplicative update algorithm. This can often produce noisy activation functions, which make the segmentation step more difficult. Some works have proposed constraining the objective function to

produce smooth activations. In this work we use the NMF algorithm presented in [14]. Here, a penalty factor is introduced to promote smoothness of H:

$$S(H) = \frac{1}{2} \sum_i \sum_n (H_i(n) - H_i(n-1))^2. \quad (7)$$

The algorithm then tries to minimize the cost function

$$D_{KL}(V|WH) + \beta S(H), \quad (8)$$

subject to the non-negativity constraints of the NMF framework. Here β is a parameter that can be used to control the smoothness of the resulting activation curves, at the expense of the algorithm having a harder time finding the appropriate components. In our experiments, a value of $\beta = 0.1$ (an order of magnitude higher than in [14]) worked well in most cases.

Another challenge with NMF is in how to select the rank, I, of the decomposition for a particular source. One approach is proposed in [15], where a singular value decomposition (SVD) is performed on V. The SVD of a matrix has the form $Z = U\Sigma V^\intercal$, where the singular values of Z lie along the diagonal of Σ. The NMF rank, I, is then estimated by finding the number of singular values that account for some proportion of the total sum along the diagonal of Σ.

From the NMF decomposition, we obtain a soft mask

$$M_i = \frac{W_i H_i}{\hat{V}} \quad (9)$$

which we can apply to the original magnitude and phase spectrograms to obtain estimates for each component, $\hat{V}_i = M_i \odot V$, $\hat{\Phi}_i = M_i \odot \Phi$ (where $\odot$ denotes the element-wise product).

3.4. Event segmentation

Segmentation is based on the observation that activations tend to loosely follow a binary on-off pattern (Figure 2). We identify sound events when the activation is above a certain threshold defined as $\mu_i + \tau_1 \sigma_i$ (where μ_i and σ_i are respectively the mean and standard deviation of the activation $H_i(n)$, and τ_1 is a paramerter) for more than 3 frames. The end of the event is then adjusted to when the activation crosses a typically lower threshold determined in the same way for a parameter τ_2. We then look for transients within the event by identifying peaks in the first order difference of $H_i(n)$, and pick them in the same way by a third threshold parameter τ_3.

When multiple transients are found within the same active event (e.g. for a rapid succession of percussive or note events with long decay), the event is split so that each transient will always start an event (although in general it is not required that all events start with a transient). A transient is defined to have a fixed number of frames corresponding to 10ms (which can be controlled as a user parameter), depending on the hop size. Finally, the 'silence' in the activation between two events is attached to the preceding event, so that an event is considered to have a transient, an active part and a silence part, where transient and silence may have zero duration. The idea is that—as the signal has a rhythmic structure—we need to proportionally scale the spaces between event onsets, but within an event we may apply different scaling.

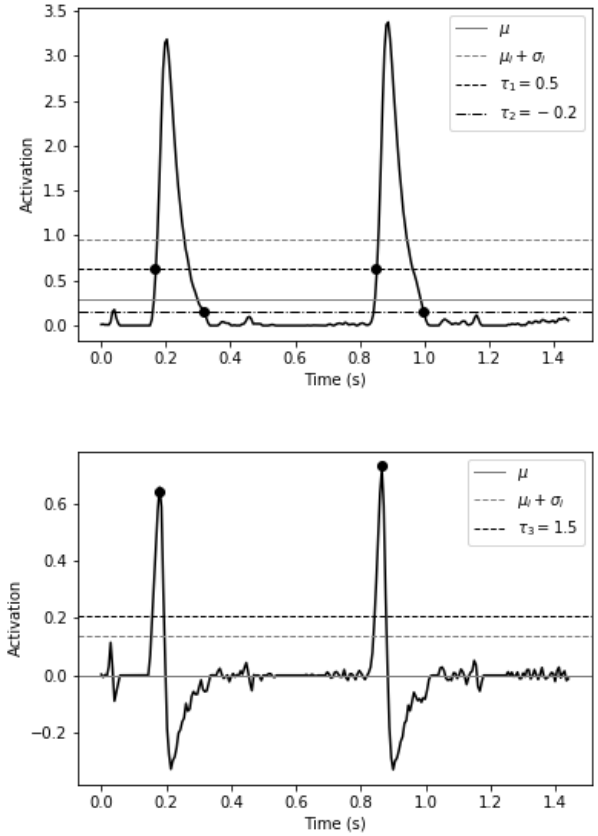

Figure 2: *Event segmentation of NMF activation. Top: activation function. Bottom: first order difference.*

3.5. Scaling and synthesis

The segmented events for each of the NMF components are then scaled according to the desired factor, r. We can take advantage of the component- and event-wise separation in several ways. First, in order to preserve the perceptual quality of transient, the transient part of each event is not scaled. By default, a new scaling factor r_A is computed for the active part. An optional feature, which can be called *envelope preservation* may be used so that the active part is also copied without scaling as if it was a transient. The silence scaling factor r_S has then to be recomputed to keep the whole of the event aligned according to r. When the event has no silence (typically as a result of splitting the event due to multiple transients) envelope preservation is not applied. Outside of transient and potentially percussive events, magnitudes are interpolated from the input $\hat{V}_i$, and phases are propagated according to the phase vocoder strategy, including identity phase locking [16]. Thus, after finding the bin k_p corresponding to the peak in the region of influence of a given magnitude bin k, we compute the phase envelope as

$$\phi_e(k,n) = \Phi(k,n) - \Phi(k_p,n), \quad (10)$$

and the deviation with respect to the bin's frequency as

$$\Delta\phi(k,n) = \Phi(k,n) - \Phi(k,n-1) - \omega(k)R \quad (11)$$

where $w(k)$ is the normalized frequency corresponding to bin k, and R is the hop size. The phase for the scaled signal in non-transient regions is then synthesized as

$$\Phi_{Yi}(k,n) = \Phi_{Yi}(k,n-1) + \omega(k)R + Arg(\Delta\phi(k,n) + \phi_e(k,n)), \quad (12)$$

where $Arg(x)$ is the principal argument function.

The estimates of the scaled spectrogram, $\hat{V}_{Yi}$ and $\hat{\Phi}_{Yi}$, are then composed into $\hat{Y}_i$ as

$$\hat{Y}_i(k,n) = \hat{V}_{Yi}(k,n)\, \mathrm{e}^{\hat{\Phi}_{Yi}(k,n)j}, \quad (13)$$

and synthesized using the inverse STFT as described in Section 3.2.

4. EXAMPLES

We implemented the proposed algorithm in a Python package, which includes a partial port of the Matlab TSM toolbox. The code is available on github[1]. While testing with several excerpts, we found that results are generally better than classic methods such as OLA or WSOLA, or the phase vocoder without transient preservation, and closer to HPSS and state-of-the-art commercial packages. A number of examples can be listened to on the companion web page for this paper[2]. It is possible to obtain good results by automating the choice of the NMF rank as outlined in Section 3.3, which often produces a large number of components. This is an interesting result, considering that the TSM is performed piecewise through potentially several hundreds of events and then reassembled, however it bears a high computational cost. In practice, a better solution is often to manually set a suitable value for the NMF rank. Generally, the algorithm works better for percussive and repetitive material, and suffers with slow frequency or amplitude modulations. With respect to the envelope preservation option, it is generally sensitive to errors in the detection of events, and hence it tends to work best for sounds that are well modeled by NMF, such as percussive loops. In these cases it can produce a more natural sound than most available algorithms. Our approach is generally comparable to using HPSS [8], which is also based on a source separation technique that decomposes the magnitude spectrogram. We now demonstrate the strengths and weaknesses of the NMF decomposition through some examples.

4.1. Glockenspiel

Using NMF tends to produce better attacks for simple percussive loops and melodies. Figure 3 shows the spectrogram of a few notes of the Glockenspiel melody included in the TSM Toolbox, as stretched by both the HPSS and the NMF approaches. It can be seen that using NMF produces sharper transient at note onsets. This is probably partly due to the NMF activations providing a good cue of the locations of transients due to individual notes, but also to the fact that our algorithm stays in the same frame rate, allowing to build a more coherent representation of the note, while the HPSS approach requires going back to the time domain for the percussive part, while staying in the frequency domain for stretching the harmonic part.

[1] https://github.com/flucoma/DAFX-2019

[2] http://www.flucoma.org/DAFX-2019/

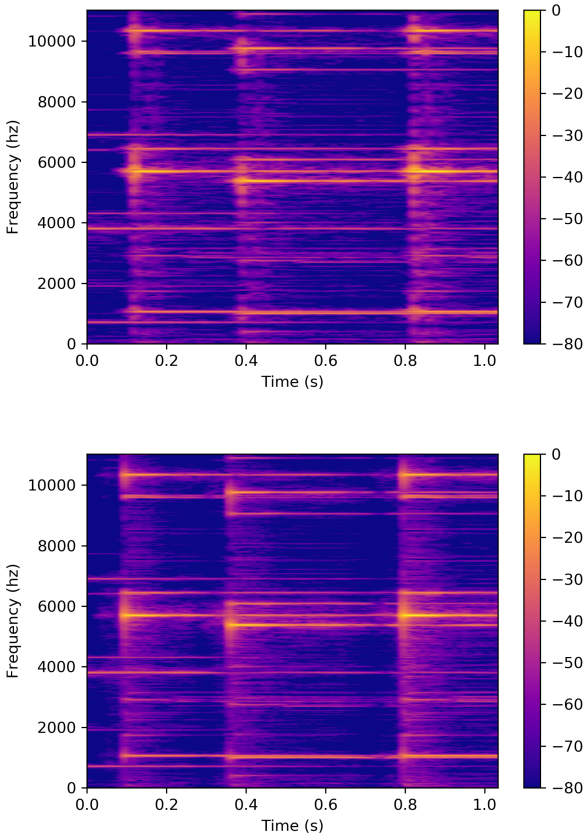

Figure 3: *Excerpt of a glockenspiel melody stretched with a 1.8 factor using HPSS (top) and NMF (bottom)*

4.2. Drum loop

One unique aspect of the proposed algorithm is the ability to modify the duration of the signal while preserving the envelope of percussive sounds. This option may introduce some artifacts if the events are not well detected, but it works well for dry percussive loops. An example is shown in Figure 4, showing the initial beats of a drum loop. Here, we can observe that the NMF approach approximates better the duration of the first two sound events (a bass drum and a hi-hat). Our approach modifies the tempo of the pattern while preserving the natural sound of each beat, while stretching via HPSS also stretches the sound's envelope, which gives it an artificial time profile. The latter is also generally the case in current commercial products.

4.3. Novel artifacts

As mentioned in the introduction, we are also interested in the creative possibilities of the failures of TSM algorithms. In this sense, we hope the proposed algorithm will also contribute new kinds of artifacts that can be used for exploring new musical possibilities. The main user parameters are the NMF rank (I), and the three parameters influencing the event segmentation (Section 3.4). Without the envelope preservation feature, our algorithm can reproduce common artifacts related with the phase vocoder. For

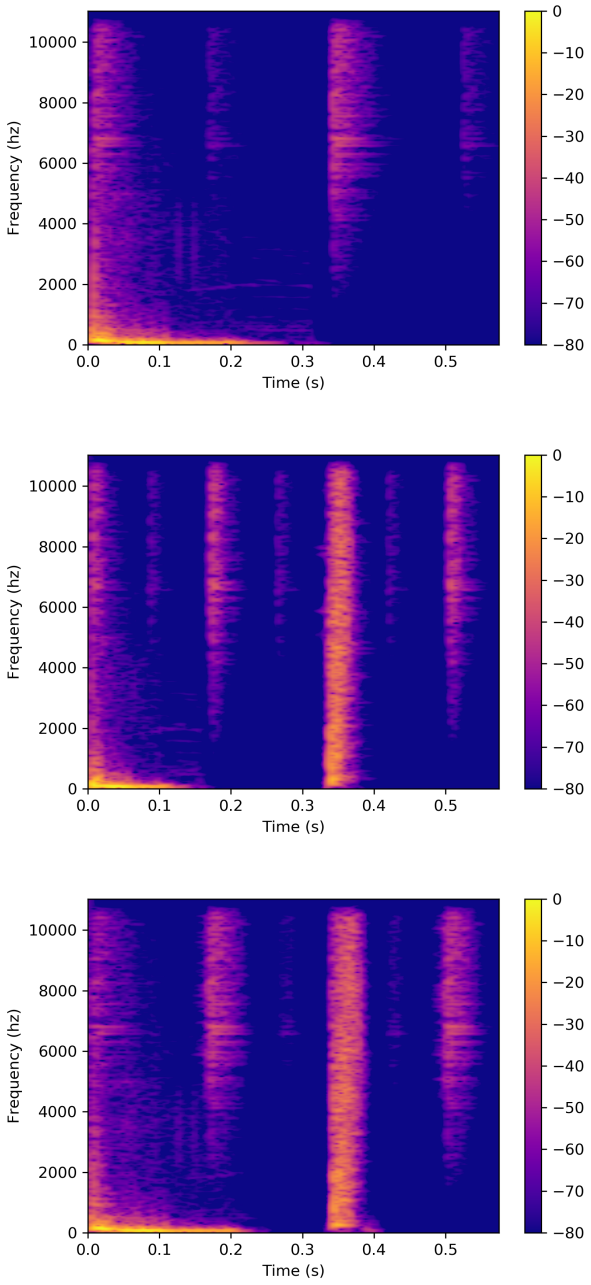

Figure 4: *Excerpt of a drum loop (top:original) stretched with a 0.6 factor using HPSS (middle) and NMF (bottom).*

example, raising the transient detection threshold τ_3 can be used to induce transient smearing, while extreme stretching factors will produce well-known phasing effects. The novel contribution of our algorithm is the possibility to preserve the envelope of a sound event, but when events are not properly detected, this results in misplaced components that produce rhythmic variations and different smearings of time not usually found in phase vocoder. An example (zoomed again for detail) is shown in Figure 5. Here, we intentionally raised the chances of mistakes by raising the rank to 20 (which is more than needed for a drum kit using mainly four sounds) and made it difficult for the algorithm to find the events by playing with parameters τ_1 and τ_2. As a result, part of the rhythm becomes confusing. The main impression is that some of the sounds have been divided and parts of them have been misplaced, creating a new rhythmic effect.

5. CONCLUSIONS AND FUTURE WORK

In this paper, we have presented an algorithm for time scale modification of audio using non-negative matrix factorization. We have presented an implementation and demonstrated several examples. The algorithm has the unique feature of being able to preserve the duration of sound events while modifying the duration of the sequence. This is generally not possible without source separation, unless the signal is purely monophonic, as the envelopes of different events tend to overlap. The NMF framework also helps generally in the identification of transients due to different components in the frequency domain. As future work, we plan to investigate strategies for synchronizing event boundaries across components so that envelope preservation can be used without compromising rhythmic structure. Similarly, classification of NMF bases would allow applying selectively to percussive events. Also, since frequency-domain TSM generally requires careful attention to the phase, we plan to experiment with complex NMF variants.

6. ACKNOWLEDGMENTS

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement n. 725899).

7. REFERENCES

[1] Paris Smaragdis and Judith C Brown, "Non-negative matrix factorization for polyphonic music transcription," in *Proceedings of the 2003 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, 2003, pp. 177–180.

[2] Jouni Paulus and Tuomas Virtanen, "Drum transcription with non-negative spectrogram factorisation," in *Proceedings of the 2005 13th European Signal Processing Conference*, 2005, pp. 1–4.

[3] Werner Verhelst and Marc Roelands, "An overlap-add technique based on waveform similarity (wsola) for high quality time-scale modification of speech," in *Proceedings of the 1993 IEEE International Conference on Acoustics, Speech, and Signal Processing*. IEEE, 1993, vol. 2, pp. 554–557.

[4] Jean Laroche and Mark Dolson, "Phase-vocoder: About this phasiness business," in *Proceedings of 1997 Workshop on Applications of Signal Processing to Audio and Acoustics*, 1997, pp. 4–pp.

[5] Jordi Bonada, "Automatic technique in frequency domain for near-lossless time-scale modification of audio.," in *Proceedings of the 2000 International Computer Music Conference*. Citeseer, 2000.

[6] Frederik Nagel and Andreas Walther, "A novel transient handling scheme for time stretching algorithms," in *Proceedings of the 127th Audio Engineering Society Convention*, 2009.

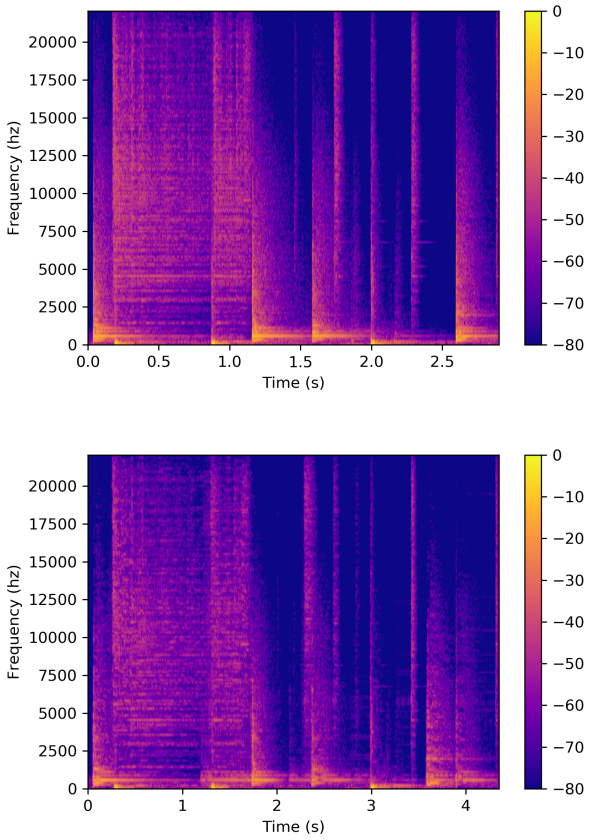

Figure 5: *Excerpt of a drum recording. Top: original. Bottom: stretched using NMF with a 1.5 factor and extreme parameters.*

[7] Shahaf Grofit and Yizhar Lavner, "Time-scale modification of audio signals using enhanced wsola with management of transients," *IEEE transactions on audio, speech, and language processing*, vol. 16, no. 1, pp. 106–115, 2008.

[8] J. Driedger, M. Müller, and S. Ewert, "Improving Time-Scale Modification of Music Signals Using Harmonic-Percussive Separation," *IEEE Signal Processing Letters*, vol. 21, no. 1, pp. 105–109, Jan. 2014.

[9] Jonathan Driedger and Meinard Müller, "TSM Toolbox: MATLAB Implementations of Time-Scale Modification Algorithms," in *Proceedings of the 2014 Conference on Digital Audio Effects (DAFX)*, 2014, p. 8.

[10] Eero-Pekka Damskägg and Vesa Välimäki, "Audio Time Stretching Using Fuzzy Classification of Spectral Bins," *Applied Sciences*, vol. 7, no. 12, pp. 1293, Dec. 2017.

[11] Marko Helen and Tuomas Virtanen, "Separation of drums from polyphonic music using non-negative matrix factorization and support vector machine," in *Proceedings of the 13th European Signal Processing Conference (EUSIPCO)*, 2005, pp. 1–4.

[12] Jordi Janer, Ricard Marxer, and Keita Arimoto, "Combining a harmonic-based nmf decomposition with transient analysis for instantaneous percussion separation," in *Proceedings of the 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2012, pp. 281–284.

[13] Daniel D Lee and H Sebastian Seung, "Algorithms for non-negative matrix factorization," in *Advances in neural information processing systems*, 2001, pp. 556–562.

[14] Slim Essid and Cédric Févotte, "Smooth nonnegative matrix factorization for unsupervised audiovisual document structuring," *IEEE Transactions on Multimedia*, vol. 15, no. 2, pp. 415–425, 2013.

[15] Hanli Qiao, "New svd based initialization strategy for non-negative matrix factorization," *Pattern Recognition Letters*, vol. 63, pp. 71–77, 2015.

[16] J. Laroche and M. Dolson, "Improved phase vocoder time-scale modification of audio," *IEEE Transactions on Speech and Audio Processing*, vol. 7, no. 3, pp. 323–332, May 1999.