MUSIC SELF-SIMILARITY MODELING USING AUGMENTED NONNEGATIVE MATRIX FACTORIZATION OF BLOCK AND STRIPE PATTERNS

Joonas Kauppinen[1]* *Anssi Klapuri*[2] *Tuomas Virtanen*[2]

[1] School of Information Sciences, University of Tampere, Finland, joonas.kauppinen@uta.fi
[2] Department of Signal Processing, Tampere University of Technology, Finland,
anssi.klapuri@tut.fi, tuomas.virtanen@tut.fi

ABSTRACT

Self-similarity matrices have been widely used to analyze the sectional form of music signals, e.g. enabling the detection of parts such as verse and chorus in popular music. Two main types of structures often appear in self-similarity matrices: rectangular blocks of high similarity and diagonal stripes off the main diagonal that represent recurrent sequences. In this paper, we introduce a novel method to model both the block and stripe-like structures in self-similarity matrices and to pull them apart from each other. The model is an extension of the nonnegative matrix factorization, for which we present multiplicative update rules based on the generalized Kullback–Leibler divergence. The modeling power of the proposed method is illustrated with examples, and we demonstrate its application to the detection of sectional boundaries in music.

Index Terms— Music structure analysis, nonnegative matrix factorization, self-similarity

1. INTRODUCTION

Music structure analysis has been an active area of research in the field of music information retrieval. Its aim is to discover the sectional form of musical works by segmenting them into series of consistent, possibly recurrent parts at a relatively large time scale. The information on sectional form can then be utilized in other tasks such as music summarization, synchronization, and visualization, as well as cover song identification [1].

Many diverse structure analysis systems have been proposed so far, some of which have been evaluated in annual MIREX evaluation campaigns [2]. Typically, the main goal of these systems is the ability to output accurate segment boundary locations together with labels that have some musical meaning comparable with those perceived by a human listener [3].

A comprehensive survey of audio-based structure analysis methods has previously been presented in [1]. Most of the methods have made use of a self-similarity matrix (SSM) constructed from a time series of acoustic feature vectors representing a song. Two main types of structures often emerge in SSMs: rectangular blocks corresponding to textures of within-parts similarities and diagonal stripes corresponding to recurrent sequences of features. The usual approach is to generate a number of candidate segment boundaries (as described in Section 2.1) and then compute the average similarity within the blocks of the SSM or employ dynamic programming to identify diagonal stripes running across those blocks.

Recently, a handful of approaches have exploited the ability of the nonnegative matrix factorization (NMF) to extract parts-based representations. For instance, the basic form of NMF has been applied to SSMs to assist in segment clustering [4]. In [5], NMF is applied to a specific score matrix combining harmonic and timbral information. Moreover, sparse, convolutive NMF has been applied to spectrogram data in order to locate recurrent harmonic motifs as well as to infer high-level structure [6].

In this paper, we introduce a model for factorizing SSMs. The model is an extension of the NMF, and its two parts model the blocks and stripes appearing in SSMs. We demonstrate that the model has potential in detecting such sectional boundaries in music that a traditional block-only approach cannot detect.

The remainder of this paper is organized as follows. Section 2 reviews some background related to SSMs and NMF. The main contribution of this paper, the NMF-based model of block and stripe structures, is presented in Section 3. Section 4 demonstrates the application of the model to the detection of sectional boundaries in music. Finally, Section 5 concludes the paper.

2. BACKGROUND

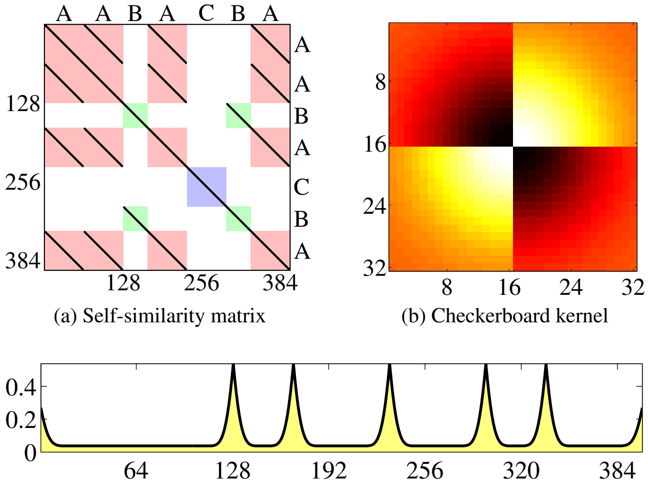
The structure in music is often immensely hierarchical. Take Western popular music, for instance: each of the high-level parts—such as verse, bridge, and chorus—tend to consist of phrases of related structure, and these phrases in turn tend to consist of more or less recurrent patterns of notes. SSMs and the NMF provide ways of visualizing and analyzing these structures.

2.1. Self-similarity matrix

Following the seminal work of Foote [7], many of the previous structure analysis methods have utilized SSMs to detect locations of recurrent patterns in songs. Standard practice is to first decompose an input audio waveform into short frames with some overlapping, and construct feature vectors that capture certain sound characteristics of songs to be analyzed. The deployed feature representations should resemble those properties of music that are known to have impact on human perception of musical structure (see [3]). Commonly used features include those related to timbre, such as the mel-frequency cepstral coefficients (MFCCs), besides those related to musical harmony (e.g. chroma), rhythm, and dynamics. An SSM (or its dual, a self-distance matrix) is then constructed by using an appropriate similarity (distance) measure to compute pairwise similarities (distances) between extracted feature vectors.

An idealized SSM is illustrated in Fig. 1a. It represents the occurences of three contrasting parts (A, B, and C) in time, and shows

*Joonas Kauppinen is funded by Tampere Doctoral Programme in Information Science and Engineering.



(a) Self-similarity matrix (b) Checkerboard kernel

(c) Novelty curve

Figure 1: Panel (a) illustrates an idealized self-similarity matrix with three contrasting parts in seven segments. Darker pixels correspond to higher similarities. Correlating the Gaussian weighted checkerboard kernel of size 32 × 32 (b) along the main diagonal of the self-similarity matrix yields the novelty curve (c).

rectangular, homogeneous blocks as well as diagonal stripes off and parallel to the main diagonal. Furthermore, a checkerboard-like structure appears along the main diagonal at sectional boundaries. Variations in tempo would show the recurrent sequences as curved stripes. However, using beat-synchronous features, i.e. computing one feature vector for each inter-beat interval, effectively straightens the stripes in SSMs.

A common, matched-filter approach to structural segmentation (first proposed in [8]) is to correlate a suitably sized, checkerboard-like kernel (Fig. 1b) along the main diagonal of an SSM. This yields a novelty curve (Fig. 1c). A peak detection algorithm can then be employed to catch the main temporal change points that in turn can be considered as candidates for sectional boundaries.

Some musical works, however, show less conspicuous block structures but several recurrent sequences. Indeed, the matched-filter approach fails to utilize the stripe information in SSMs and thus also fails to recognize boundaries between successive repetitions of a segment. For example, in the case of Fig. 1, the boundary between the two A parts at time 64 is not reflected by the novelty curve.

2.2. Nonnegative matrix factorization

The NMF aims at approximating a nonnegative matrix by a product of two lower-rank matrices. First introduced by Lee and Seung [9], the NMF and its various extensions have recently been applied to numerous problems in image processing, microarray analysis, sound recognition, and text mining to name but a few [10].

Let $\boldsymbol{S}$ denote a matrix of size $I \times J$ with nonnegative elements $\boldsymbol{S}_{i,j}$. Then, the NMF of $\boldsymbol{S}$ is given by $\boldsymbol{S} \approx \boldsymbol{V} = \boldsymbol{W}\boldsymbol{H}$, where $\boldsymbol{W}$ and $\boldsymbol{H}$ are nonnegative basis and gain matrices of sizes $I \times K$ and $K \times J$, respectively. Scalar K denotes the rank of the approximation, i.e. number of components. Matrices $\boldsymbol{W}$ and $\boldsymbol{H}$ are chosen so that their product $\boldsymbol{V}$ gives an optimal approximation to $\boldsymbol{S}$ w.r.t. given cost function. Commonly used cost functions include the sum of squared Euclidean distances and the generalized Kullback–Leibler (KL) divergence, the latter of which is defined as

$$\mathrm{KL}\left(\boldsymbol{S}, \boldsymbol{V}\right) := \sum_{i,j} \left(\boldsymbol{S}_{i,j} \log \frac{\boldsymbol{S}_{i,j}}{\boldsymbol{V}_{i,j}} - \boldsymbol{S}_{i,j} + \boldsymbol{V}_{i,j} \right). \tag{1}$$

3. PROPOSED MODEL

In this section, we propose to factorize SSMs using an augmented NMF that is specifically tailored to represent the block and stripe structures in SSMs.

3.1. Block structures

Let $\boldsymbol{S}$ denote an SSM of size $I \times I$. Since $\boldsymbol{S}$ is symmetric, we propose to approximate $\boldsymbol{S}$ by the structured, symmetric NMF [11]:

$$\boldsymbol{S}_{i,j} \approx \boldsymbol{V}_{i,j} = \sum_{k=1}^{K} \boldsymbol{A}_{i,k} \boldsymbol{B}_{k,k} \boldsymbol{A}_{j,k}, \tag{2}$$

where $\boldsymbol{A}$ is $I \times K$ and $\boldsymbol{B}$ is diagonal $K \times K$. Each matrix has nonnegative elements only. The partial derivatives of (1) with respect to elements $\boldsymbol{A}_{i,k}$ and $\boldsymbol{B}_{k,k}$ lead to multiplicative update rules

$$\boldsymbol{A}_{i,k} \leftarrow \boldsymbol{A}_{i,k} \frac{\sum_j \boldsymbol{R}_{i,j} \boldsymbol{A}_{j,k}}{\sum_{i,j} \boldsymbol{R}_{i,j} \boldsymbol{A}_{i,k} \boldsymbol{A}_{j,k}}, \tag{3}$$

$$\boldsymbol{B}_{k,k} \leftarrow \boldsymbol{B}_{k,k} \sum_{i,j} \boldsymbol{R}_{i,j} \boldsymbol{A}_{i,k} \boldsymbol{A}_{j,k}, \tag{4}$$

where $\boldsymbol{R}_{i,j} = \boldsymbol{S}_{i,j} / \boldsymbol{V}_{i,j}$. The KL divergence is nonincreasing under these rules.

3.2. Stripe structures

We propose to extend (2) with a term capable of representing stripe structures:

$$\boldsymbol{S}_{i,j} \approx \boldsymbol{V}_{i,j} = \sum_{k=1}^{K} \boldsymbol{A}_{i,k} \boldsymbol{B}_{k,k} \boldsymbol{A}_{j,k} + \sum_{\ell=1}^{L} \boldsymbol{C}_{i-j+1,\ell} \boldsymbol{D}_{\ell,i+j-1}, \tag{5}$$

where $\boldsymbol{A}$ and $\boldsymbol{B}$ are as in (2), $i \geq j$, and the nonnegative matrices $\boldsymbol{C}$ and $\boldsymbol{D}$ are $I \times L$ and $L \times (2I - 1)$, respectively. Rules (3) and (4) clearly hold for minimizing (1) for (5). For the latter part of (5), we index the rows of $\boldsymbol{C}$ by $\tau := i - j + 1$ and columns of $\boldsymbol{D}$ by $t := i + j - 1$. This corresponds to modeling $\boldsymbol{S}$ in a 45-degrees-rotated fashion as illustrated in Fig. 2.

Following the approach of [12], we obtain new estimates for $\boldsymbol{C}_{\tau,\ell}$ and $\boldsymbol{D}_{\ell,t}$ by multiplying the previous estimate by the ratio of negative and positive terms of the corresponding partial derivative. We get the following update rules:

$$\boldsymbol{C}_{\tau,\ell} \leftarrow \boldsymbol{C}_{\tau,\ell} \frac{\sum_{i=\tau}^{I} p\left(\tau\right) \cdot \boldsymbol{R}_{i,i-\tau+1} \boldsymbol{D}_{\ell,2i-\tau}}{\sum_{i=\tau}^{I} p\left(\tau\right) \cdot \boldsymbol{D}_{\ell,2i-\tau}}, \tag{6}$$

$$\boldsymbol{D}_{\ell,t} \leftarrow \boldsymbol{D}_{\ell,t} \frac{\sum_{i=\lfloor t/2 \rfloor+1}^{\min(t,I)} q\left(i,t\right) \cdot \boldsymbol{R}_{i,t-i+1} \boldsymbol{C}_{2i-t,\ell}}{\sum_{i=\lfloor t/2 \rfloor+1}^{\min(t,I)} q\left(i,t\right) \cdot \boldsymbol{C}_{2i-t,\ell}}, \tag{7}$$

where $\lfloor \cdot \rfloor$ denotes the floor function, and functions p and q are defined as

$$p\left(\tau\right) := \begin{cases} 1 & \text{if } \tau = 1, \\ 2 & \text{if } \tau \neq 1, \end{cases} \qquad q\left(i,t\right) := \begin{cases} 1 & \text{if } t = 2i - 1, \\ 2 & \text{otherwise.} \end{cases} \tag{8}$$

The multipliers p and q are needed because only the elements corresponding to the lower triangular part of $\boldsymbol{S}$ are updated.

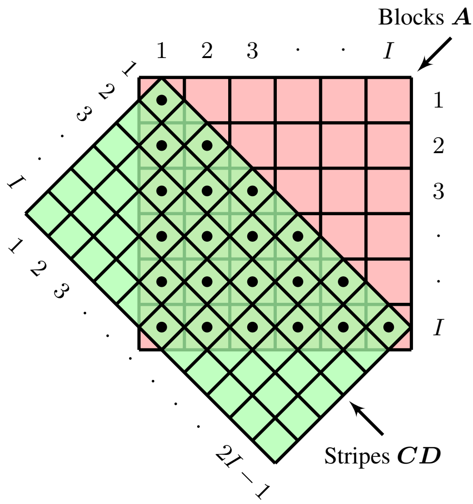


Figure 2: Illustration of the block and stripe parts of the proposed model. Only elements marked with a dot are utilized for estimating the stripe part of the model.

Algorithm 1 Estimate $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$, and $\boldsymbol{D}$ in (5)

```
initialize $A_{i,k}$, $B_{k,k}$, $C_{\tau,\ell}$, and $D_{\ell,t}$ to random positive values for all $i$, $k$, $\tau$, $\ell$, and $t$
normalize $A$ and $B$ such that $\sum_i A_{i,k} = 1$ for all $k$ and $\sum_k B_{k,k} = \sum_{i,j} S_{i,j}$ as proposed in [11]
compute $V$ using (5) and set $R_{i,j} \leftarrow S_{i,j}/V_{i,j}$ for all $i$ and $j$
repeat
    update $B$, $A$, $D$, and $C$ using (4), (3), (7), and (6), respectively, ensuring no element is set to exactly zero; after each of these, update $V$ using (5) and set $R_{i,j} \leftarrow S_{i,j}/V_{i,j}$ for all $i$ and $j$
until convergence
return $A$, $B$, $C$, $D$
```

3.3. Algorithm

Algorithm 1 gives an overview of estimating the matrices in (5). Choosing suitable convergence criteria and numbers of components K and L is beyond the scope of this paper. Hence, we will use fixed numbers of components and iterations in the following Section 4. In a more sophisticated approach, a relatively large number of components could be estimated, clustering similar components together at a later stage. For our model, the numbers of clustered components $K_\star$ and $L_\star$ should roughly correspond to the number of musical parts and the number of parallel stripes in the lower triangular part of $\boldsymbol{S}$, respectively.

4. CASE STUDY

We demonstrate a system for detecting sectional boundaries in music based on the proposed model. Fig. 3 shows an overview of the system. In this paper, we use The Beatles song "All My Loving" as an example.

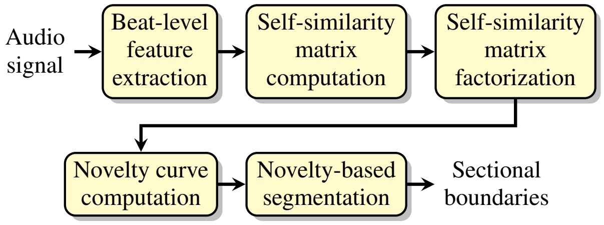
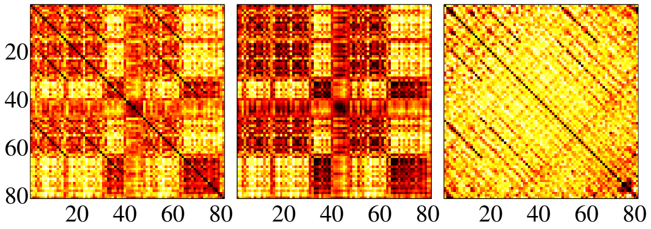

Audio signal
Beat-level feature extraction
Self-similarity matrix computation
Self-similarity matrix factorization
Novelty curve computation
Novelty-based segmentation
Sectional boundaries

Figure 3: Overview of the proposed boundary detection system. See Section 4 for further details.



(a) Original SSM (b) Block structures (c) Stripe structures

Figure 4: Self-similarity matrices of The Beatles song "All My Loving" computed using beat-synchronous features. Darker pixels correspond to higher similarities. The approximation of (a) can be expressed as the sum of the estimated block structures (b) and stripe structures (c).

4.1. Beat-level feature extraction

As the first task, we extract MFCCs and chroma to represent the timbral and harmonic content of the input audio signal using the MATLAB toolbox described in [13]. These features are computed in $2^{13}/44100 \approx 0.186$-second frames with 75% overlapping. For the chroma, the tuning is estimated beforehand by applying the method detailed in [14] with $1/100$ semitone precision. The obtained features are averaged over beat frames after applying the beat tracking algorithm of [15]. For the purpose of illustrations, we perform the averaging over frames of 4 beats.

Finally, each element of the beat-synchronous MFCC vectors is standardized to mean zero and unit variance over the song under investigation. The chroma are nonnegative and are not normalized as they represent the energy associated with each of the 12 pitch classes: scaling to, say, unit sum would put more weight to occurences of rare pitch classes compared to frequent ones.

4.2. Self-similarity matrix computation

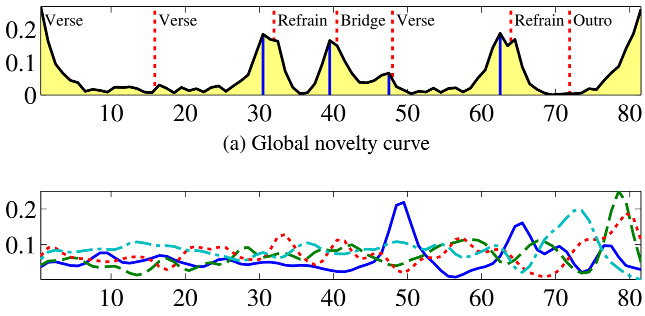
Let $\boldsymbol{f}_i$ and $\boldsymbol{g}_i$ denote the feature vectors containing MFCCs and chroma, respectively, at time point i. Based on the cosine of the angle between two vectors, we use the following metric to represent the similarity of features between time points i and j:

$$\boldsymbol{S}_{i,j} := \left(\frac{1}{2} + \frac{\boldsymbol{f}_i' \boldsymbol{f}_j}{2\,\|\boldsymbol{f}_i\|\,\|\boldsymbol{f}_j\|} + \frac{\boldsymbol{g}_i' \boldsymbol{g}_j}{\|\boldsymbol{g}_i\|\,\|\boldsymbol{g}_j\|} \right) \Big/ 2, \qquad (9)$$

where $\|\cdot\|$ denotes the Euclidean (L2) norm. Metric (9) corresponds to the average of two similarity measures, one for MFCCs and one for the chroma, each of which can have values between zero and one. Computing (9) for all tuples (i, j) yields the SSM (Fig. 4a).

	$L = 0$	$L = 4$	$L = 8$	$L = 12$	$L = 16$
$K = 0$	—	42.4	22.2	11.1	5.3
$K = 2$	37.6	19.0	10.4	5.3	2.4
$K = 4$	14.2	7.3	4.4	2.8	1.8
$K = 6$	5.6	3.0	2.2	1.6	1.0
$K = 8$	2.7	1.8	1.4	1.0	0.8

Table 1: KL divergences between the original self-similarity matrix of the example song (Fig. 4a) and approximations obtained using different numbers of components K and L.



(a) Global novelty curve

(b) Basis components

Figure 5: Panel (a) illustrates a global novelty curve as a function of time (see Section 4.4). Ground truth locations of sectional boundaries are shown as dashed lines. Panel (b) shows the estimated basis components.

4.3. Self-similarity matrix factorization

Running the Algorithm 1 with $K = 4$ and $L = 4$ yields Fig. 4b and Fig. 4c displaying the estimated block and stripe structures of Fig. 4a. We observe that the algorithm does well in estimating the lags between recurrent patterns but fails to recognize clear start and end points of them. In addition, some noise can be seen at the corners of Fig. 4c due to the smaller amount of data.

Table 1 illustrates the behavior of the KL divergence values after 1,000 iterations as a function of K and L. To obtain these values, same initial values where used for the factor matrices. As expected, the stripe components seem to model a relatively small area of the SSM under investigation.

4.4. Novelty curve computation and segmentation

We compute a global novelty curve by applying a checkerboard kernel (see Fig. 1) on the reconstruction of the $\boldsymbol{ABA'}$ part of the model. We pick n largest peaks from the curve with the constraint that the minimum distance between two consecutive peaks is 8 seconds. The curve and peaks are shown in Fig. 5a.

Fig. 5b illustrates the basis components in $\boldsymbol{C}$. A Gaussian kernel has been convoluted along them. The peaks correspond to lags between recurrent patterns. The exact mechanism of using these components for segmentation is beyond the scope of this paper.

5. CONCLUSIONS

We have introduced an NMF-based model that estimates block and stripe structures in SSMs simultaneously. Multiplicative update rules based on the KL divergence were presented. In addition, we demonstrated the potential the model has for improving the performance of sectional boundary detection.

On the one hand, we have observed that the stripe part of the model is able to pick the lags between recurrent patterns adequately. On the other hand, we have noticed that is is not able to produce clear structures in gains of the estimated stripes. Therefore, we are currently investigating several variants that exploit time-lag matrices corresponding to SSMs. We plan to inquire into these models with additional sparsity constraints and to perform extensive evaluations with annotated music collections spanning several genres.

6. REFERENCES

[1] J. Paulus, M. Müller, and A. Klapuri, "Audio-based music structure analysis," in *Proc. ISMIR*, 2010, pp. 625–636.

[2] A. F. Ehmann, M. Bay, J. S. Downie, I. Fujinaga, and D. De Roure, "Music structure segmentation algorithm evaluation: Expanding on MIREX 2010 analyses and datasets," in *Proc. ISMIR*, 2011, pp. 561–566.

[3] M. J. Bruderer, "Perception and modeling of segment boundaries in popular music," Ph.D. dissertation, Technische Universiteit Eindhoven, Eindhoven, Netherlands, 2008.

[4] F. Kaiser and T. Sikora, "Music structure discovery in popular music using non-negative matrix factorization," in *Proc. ISMIR*, 2010, pp. 429–434.

[5] R. Chen and M. Li, "Music structural segmentation by combining harmonic and timbral information," in *Proc. ISMIR*, 2011, pp. 477–482.

[6] R. J. Weiss and J. P. Bello, "Unsupervised discovery of temporal structure in music," *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 6, pp. 1240–1251, 2011.

[7] J. Foote, "Visualizing music and audio using self-similarity," in *Proc. ACM Multimedia*, 1999, pp. 77–80.

[8] ——, "Automatic audio segmentation using a measure of audio novelty," in *Proc. ICME*, vol. 1, 2000, pp. 452–455.

[9] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, no. 6755, pp. 788–791, 1999.

[10] A. Cichocki, R. Zdunek, A. H. Phan, and S.-i. Amari, *Nonnegative Matrix and Tensor Factorizations: Applications to Exploratory Multi-way Data Analysis and Blind Source Separation*. Chichester, UK: Wiley, 2009.

[11] B. Vanluyten, J. C. Willems, and B. De Moor, "Structured nonnegative matrix factorization with applications to hidden Markov realization and clustering," *Linear Algebra Appl.*, vol. 429, no. 7, pp. 1409–1424, 2008.

[12] T. Virtanen, "Monaural sound source separation by nonnegative matrix factorization with temporal continuity and sparseness criteria," *IEEE Audio, Speech, Language Process.*, vol. 15, no. 3, pp. 1066–1074, 2007.

[13] O. Lartillot and P. Toiviainen, "A MATLAB toolbox for musical feature extraction from audio," in *Proc. DAFx*, 2007, pp. 237–244.

[14] M. Müller and S. Ewert, "Chroma toolbox: MATLAB implementations for extracting variants of chroma-based audio features," in *Proc. ISMIR*, 2011, pp. 215–220.

[15] D. P. W. Ellis, "Beat tracking by dynamic programming," *J. New Music Res.*, vol. 36, no. 1, pp. 51–60, 2007.