AUDIO EVENT DETECTION BASED ON LAYERED SYMBOLIC SEQUENCE REPRESENTATIONS

Michele Lai Chin *Juan José Burred*

Audionamix
114, avenue de Flandre, 75019 Paris, France
{michele.lai.chin,juan.jose.burred}@audionamix.com

ABSTRACT

We introduce a novel application of genetic motif discovery in symbolic sequence representations of sound for audio event detection. Sounds are represented as a set of parallel symbolic sequences, each symbol representing a spectral shape, and each layer indicating the contribution weights of each spectral shape to the sound. Such layered symbolic representations are input to a genetic motif discovery algorithm that detects and clusters recurrent and structurally salient sound events in an unsupervised and queryless manner. The found motifs can be interpreted as statistical temporal models of spectral evolution. The system is successfully evaluated in two tasks: environmental sound event detection, and drum onset detection.

Index Terms— Audio event detection, motif discovery, symbolic representations.

1. INTRODUCTION

The goal of this work is to take advantage of genetic *motif discovery* algorithms' speed, accuracy and flexibility to perform audio event detection. Motif discovery is a common task in bioinformatics; it reveals the biological significance of a portion of genetic code [1]. It is performed on very large data, usually DNA sequences of several millions of symbols. Existing methods are therefore designed to be extremely fast. Furthermore, motif repetitions are seldom exact in genetics, and thus retrieved matches need to be detected based on a carefully designed measure of similarity; the same can be said about realistic sound events.

Motif discovery is related to *sequence alignment*, with an important difference: sequence alignment takes a query sequence and finds the positions of best alignments in a database, while motif discovery finds relevant events from scratch and creates a statistical model of the event (the motif). Thus, motif discovery is *data-driven* (representations are directly learned from the sound on which event detection is performed), *unsupervised* (there is no a priori learning) and *queryless* (the algorithm decides by itself which are the interesting similarity regions). While genetic sequence alignment has often been used in audio applications before (see e.g. [2]), to our knowledge the use of genetic motif finding in audio is new.

In our case, each sequence letter corresponds to a characteristic spectral shape, and thus a sequence motif can be interpreted as a statistical temporal model of the spectrum. In a parallel work [3], we have proposed a baseline system that generates a single symbolic sequence for each sound and subjects it to motif discovery. Here, we extend that idea and propose a system that translates each file into a set of parallel symbolic sequences, which we call a *layered sequence representation*. Instead of representing each spectral frame by a single symbol at a time, each frame can now be described by a combination of symbols, obtaining different levels of representation. This is useful for describing complex sounds.

This work is supported by the OSEO-funded EUREKA Eurostars project RAABSPM (AudioHelix), E!5189.

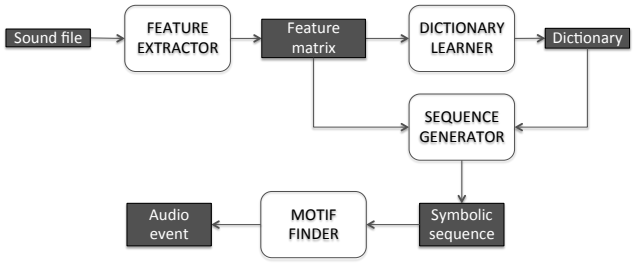

Fig. 1. Overview of the audio event detection system.

Since traditional genetic motif discovery is constrained to single-level sequences, several important adaptations to our application scenario are needed. In particular, this paper focuses on two main issues: first, the study and comparison of two methods for learning layered symbolic representations, based on Non-negative Matrix Factorization (NMF) and Principal Component Analysis (PCA); and second, the influence of correlations between layered sequences, by taking cross-layer structure into account. As motif discovery algorithm, we have chosen the well-established Multiple Expectation-Maximization for Motif Elicitation (MEME) [4]. We evaluate the system with an artificial mixture of environmental sounds with two sound event classes (dog barks and hammer strikes), and with a collection of music excerpts where the goal is to detect the onsets of the bass drum and the snare drum.

2. SYSTEM OVERVIEW

Our framework for audio event detection in a sound excerpt relies on four steps, as illustrated in Fig. 1. First, audio features are calculated for every overlapping time window of the excerpt. We use 10 ms-long windows and 50% overlapping, and computed 12 Mel-Frequency Cepstral Coefficients (MFCC), the first coefficient (energy) being discarded. We obtain a set of feature vectors ordered in time.

Second, the learning algorithms compute a set of prototype feature vectors that we call a *dictionary*, and give a representation of the feature vectors as combinations of the dictionary elements, called *atoms*. Here, we will discuss the advantages and drawbacks of

two dictionary learning methods: one based on NMF, which views data feature vectors as sums of nonnegative dictionary elements, and PCA, which relies on variance maximization.

Third, a symbol (usually a letter from the alphabet) is assigned to every dictionary element. Hence symbolic time sequences are generated to represent the sound excerpt as a series of symbols. We will also discuss the issue of how these sequences can be built to help audio event detection. Finally, the MEME algorithm is run on the symbolic sequence, and the resulting sequence motifs are mapped to the corresponding audio events.

2.1. Dictionary learning algorithms

As previously stated, we tested two well-known methods for learning the dictionary: NMF and PCA. NMF formulates the matrix factorization problem under non-negativity constraints, and is implemented as an iterative optimization with a given objective function. Given a nonnegative data matrix $\mathbf{X}$, we look for a nonnegative dictionary matrix $\mathbf{W}$ and activations matrix $\mathbf{H}$ so that $\mathbf{X} \approx \mathbf{WH}$, with $\mathbf{W}$ and $\mathbf{H}$ smaller than $\mathbf{X}$. In our case, $\mathbf{X}$'s columns are the data feature vectors, $\mathbf{W}$'s columns are the atoms and $\mathbf{H}$ contains the activation coefficients, i.e., the weights with which the atoms need to be combined to reconstruct the original signal. We will denote the number of atoms by K. We iteratively minimize the difference between $\mathbf{X}$ and $\mathbf{WH}$ according to the Itakura-Saito divergence.

PCA can also be formulated as a matrix factorization, but the constraints are different. PCA searches a new coordinate system concentrating the information in the subspace spanned by its first vectors. Thus, in the PCA case, $\mathbf{W}$'s columns (atoms) are the maximum-variance directions in the new space, which are equal to the K eigenvectors of the covariance matrix of the data which correspond to the K larger eigenvalues. $\mathbf{H}$ contains again the activation coefficients, but now elements in both $\mathbf{W}$ and $\mathbf{H}$ are allowed to be negative.

Once the dictionary $\mathbf{W}$ of K atoms is constructed either way, each atom is assigned to a letter from an alphabet of size K. The choice and assignment of letters is arbitrary. The feature vector sequence is thus converted to a sequence of letters, according to the atoms having the highest weights at each frame. Note that the use of NMF or PCA on non-additive features such a MFCC poses several theoretical questions, which will be discussed in Sect. 3.1.

2.2. Motif discovery with MEME

MEME [4] is an algorithm commonly used for motif search in genetic sequences. It relies on a two-component mixture model: a portion of sequence can either be a *motif* occurrence or part of the *background*, according to a binomial distribution. If a portion of sequence is a motif, each one of its symbols is generated by a multinomial distribution (containing the probability of appearance of each symbol) specific to that particular position in the motif. If it belongs to the background, its symbols are generated by the same multinomial distribution, independently of the position. The distribution for a given substring $\mathbf{x}_i$ can thus be expressed as

$$p(\mathbf{x}_i|\boldsymbol{\theta}) = \lambda p_M(\mathbf{x}_i|\boldsymbol{\theta}_M, w) + (1-\lambda)p_B(\mathbf{x}_i|\boldsymbol{\theta}_B), \qquad (1)$$

where $\boldsymbol{\theta}_M$ is the parameter vector of the set of multinomials describing the motif, $\boldsymbol{\theta}_B$ is the parameter vector of the single multinomial describing the background, λ is the probability that the substring was generated by the motif model, and $\boldsymbol{\theta}$ is the global parameter vector, defined as $\boldsymbol{\theta} = \{\lambda, \boldsymbol{\theta}_M, \boldsymbol{\theta}_B\}$. The index w indicates that $\boldsymbol{\theta}_M$ depends on the position of the symbols in the motif.

The parameters are subjected to Maximum Likelihood estimation via an Expectation-Maximization algorithm. In addition, a series of heuristics are performed to find good candidates for the starting positions of the motif occurrences. Details about the actual computation are to be found in [4]. The algorithm outputs, for each found motif, a Position-Specific Probability Matrix (PSPM), containing the probability of each letter at each position in the motif, and a list of the starting and end points of each instance of the motif in the sequence database. Such motif instances are called *sites*.

3. LAYERED SYMBOLIC SEQUENCES

3.1. Interpretation of dictionary elements

NMF and PCA are both matrix factorization techniques, in that they express the data items as linear combinations of dictionary elements. The coefficients are usually called activations for NMF and scores for PCA; we call them both *weights*, and regard them as a measure of importance of dictionary elements in the mixture. The larger the weight is, the more important the corresponding component is. However, the application of NMF and PCA to an MFCC feature space raises several questions, which are worth discussing here.

In the case of NMF, the data is viewed as a sum of dictionary elements, with a nonnegativity constraint. But note that audio features are not always nonnegative, and that the chosen audio features need to be additive for this decomposition to be intuitive or physically interpretable. This suggests the use of features such as the magnitude spectrum, or filter banks. Here, in contrast, we are using NMF on a non-additive representation (MFCC). This means that, even if the observed feature vectors can be indeed reconstructed by summing the dictionary elements, the individual atoms will not necessarily correspond to the individual sound entities in the spectrogram domain. Furthermore, MFCCs contain negative values, and thus they need to be positivized before the NMF computation.

On the other hand, using PCA on MFCCs can seem redundant, since the last stage of the MFCC extraction process involves computing a Discrete Cosine Transform (DCT), which already has good (while non-optimal) data compression properties.

Finally, in the case of PCA, weights may be negative. In terms of spectral contents, negative weights do not bring a principal component into the mix, but its inverse. Here, we choose the usual order of numbers to sort components by importance according to their PCA score. In future work, it might be interesting to consider negative scores as dual cases, and define additional atoms as the inverses of existing ones.

Due to these theoretical considerations, we performed a set of preliminary experiments to study the effect of applying NMF and PCA to power spectrograms and Mel filter banks (i.e., MFCCs without the logarithm and DCT stages) instead of MFCCs. For NMF, the MFCCs were positivized simply by adding the minimum value in the feature matrix. Surprisingly, both methods worked better with MFCCs than with spectrograms or filter banks. Hence, both factorizations seem to work well as dimensionality reduction stages in this particular application scenario. However, to draw more solid conclusions on the implications of this way of factorizing MFCCs, further experiments and evaluation tasks would be needed.

3.2. Independent layered sequences

Let us now define an N-sequence representation, where N must be smaller than the data representation space dimension K. Sequence no. 1 is the series of symbols corresponding to the most relevant

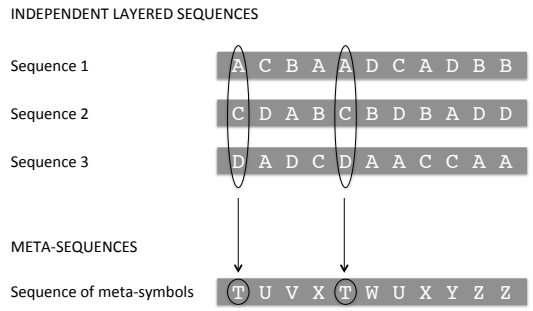

Fig. 2. Layered sequence and metasymbol representations.

dictionary component in each one of the consecutive overlapping analysis windows. Relevance is measured by the NMF or PCA decomposition weights, as defined before. Similarly, sequence no. n is the series of symbols corresponding to the n^{th} most relevant dictionary component in each one of the consecutive overlapping analysis windows. This is what we call a layered sequence representation. To achieve better performance, before building the sequences, we discard components whose weights are too low by thresholding. Thus, there may be less than N relevant components per sequence position; we authorize blank symbols.

The particular case $N = 1$ corresponds to generating a single-level sequence, as in [3]. In that case, the generated sequences can be treated by MEME without further processing. For $1 < N \leq K$, we have a layered sequence, and MEME has to be accordingly adapted. In particular, we would need to add an extra dimensionality N to model parameters $\boldsymbol{\theta}_M$ and $\boldsymbol{\theta}_B$ to account for the extra layers. However, if we assume the N layered symbols to be independent, it can be shown that such a layered MEME approach is equivalent to inputting layered symbols sequentially into standard MEME. This requires however to perform motif search in a representation that is N times longer.

3.3. Metasymbols and structured layered sequences

The previous approach is based on the assumption that layered sequences are statistically independent. If we want to model a certain degree of structure among sequences, and still be able to perform the motif search with MEME, we need a higher level representation taking into account cross-layer dependencies.

Given the alphabet of K symbols defined by dictionary learning, and assuming that we have N layered sequences ($N \leq K$), we define a new alphabet of metasymbols, which are all the combinations of K or less symbols, without repetition, but with permutations. The size of such meta-alphabet is given by the sum:

$$K_{\text{meta}} = \sum_{k=1}^{K} (K)_k, \tag{2}$$

where $(K)_k$ is the Pochhammer symbol denoting a falling factorial:

$$(K)_k = K(K-1)(K-2)\dots(K-k+1). \tag{3}$$

Note that this number grows extremely fast. For $K = 3$, we have $K_{\text{meta}} = 15$ metasymbols, but $K = 5$ gives already $K_{\text{meta}} = 325$ metasymbols. Yet this simple approach effectively models structure between sequences at all given positions for a moderate number of metasymbols, as experiments will show, and does not require the symbolic sequence to be extended as in the previous case. The formation of a metasymbolic sequence from a layered sequence is illustrated in Fig. 2.

To sum up, we have implemented and tested these three sequence generation methods:

- **Single-layer sequences.** At each frame, the atom with the highest weight is chosen from an alphabet of K atoms. For single-layer sequences, we will use the notation $K/1$.
- **Independent layered sequences.** The $N > 1$ most important atoms are chosen from an alphabet of K atoms. This will be denoted by the notation K/N.
- **Meta-sequences.** From an alphabet of K atoms and $K = N$ sequences, a sequence of K_{meta} metasymbols is generated. This will be denoted by the notation $K/1/m(K_{\text{meta}})$.

4. APPLICATION TO EVENT DETECTION

In the symbolic representation of an audio excerpt, we expect recurrent motifs to correspond to structurally relevant audio events. Such events can be of any type, and we characterize them by a certain temporal evolution of the audio features we extract from them. An audio event is spotted by looking for a particular, recurrent series of audio feature characteristics. Depending on the chosen feature set, it might denote different things, from sound volume to spectral or cepstral profiles (our case).

To evaluate the system, we use a collection of sound mixes where the onsets of repetitive short audio events have been manually annotated. MEME outputs, for each found motif, the list of found sites in terms of sequence indices, which are translated to time and matched with the annotated ground truth. Note that, by grouping the sites with the motifs, MEME is concurrently performing onset detection and unsupervised classification (clustering) of the audio events, two tasks that are traditionally separated in audio analysis or Music Information Retrieval (MIR) tasks.

The performance is measured in terms of class-wise Precision (P), Recall (R) and F-Measure $F = (2PR/(P+R))$ with respect to the annotated onsets within an error window of 40 ms. F is considered the overall quality indicator. The system was tested in two application scenarios: detection of events in an environmental sound scene, and bass drum and snare drum detection in music mixtures.

4.1. Results for environmental sound event detection

For the first test, an artificial environmental sound scene was created by mixing a background sound of a park ambiance with repetitive (but non-identical) dog barks and hammer strikes. NMF and PCA were compared as dictionary learning methods, with different configurations for sequence generation. In particular, we chose the 3/1 and 12/1 configurations for single-layer sequences, the 3/3 and 12/3 configurations for the independent layered sequences and the 3/1/m(15) configuration for the metasequence. We chose $K = 12$ in two of the configurations because it is the highest possible alphabet size due to the maximum feature dimensionality of 12 MFCCs. The choice of $K = 3$ in the three other configurations was to compare the gain of using metasequences, while keeping a reasonable number of metasymbols ($K_{\text{meta}} = 15$).

The results are shown in Fig. 3, where we have measured F for different levels of the background ambiance sound (measured as the peak-to-peak amplitude ratio in dB). The rightmost point in the graph corresponds to an infinite level ratio (no background sound).

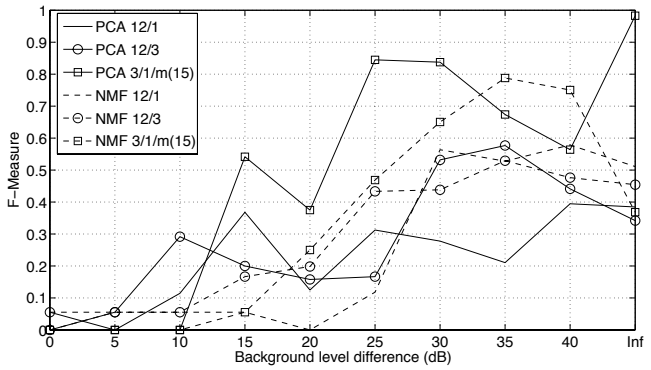

Fig. 3. Results (F-Measure) for environmental sound detection.

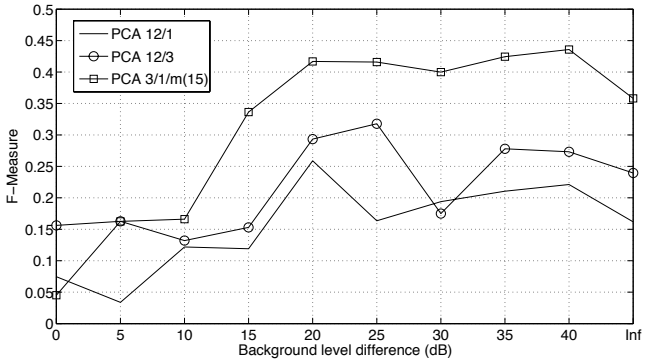

Fig. 4. Results (F-Measure) for drum sound detection.

	Environmental			Drums		
Configuration	P	R	F	P	R	F
NMF 3/1	30.43	46.67	36.84	13.63	20.24	14.76
NMF 3/3	16.65	56.25	25.56	12.47	33.09	13.69
NMF 12/1	22.92	9.58	11.81	16.47	16.87	13.23
NMF 12/3	41.67	45.83	43.33	12.42	38.29	17.78
NMF 3/1/m(15)	44.12	50.00	46.87	20.53	28.72	20.87
PCA 3/1	38.10	22.08	25.56	9.27	8.21	8.00
PCA 3/3	53.12	35.42	30.36	11.46	57.06	18.52
PCA 12/1	31.25	31.25	31.25	21.35	22.03	16.35
PCA 12/3	50.00	10.00	16.67	30.57	53.79	31.78
PCA 3/1/m(15)	89.58	80.42	**84.49**	44.58	44.37	**41.59**

Table 1. Full results at 25 dB background level difference.

For most background levels, the highest performing configuration is PCA 3/1/m(15), reaching $F = 84.59\%$ for a level of 25 dB and of 98.28% with no background noise. It can be seen that overall, PCA performs better than NMF, and that the gain of using metasequences over independent layered sequences, and even more over single-layer sequences, is considerable. More detailed results for the case of 25 dB background level ratio are shown on the left part of Table 1. With PCA, using metasymbols with the same original dictionary size of $K = 3$ improves F by 58.93%, while increasing the dictionary size from $K = 3$ to $K = 12$ only results in a moderate improvement of 5.59%. The results with NMF are more similar among configurations.

4.2. Results for drum sound detection

The same configurations have been tested in a drum detection task. A set of 5 monaural mixes, of 10 seconds each, was created by mixing real music excerpts where the drum tracks were available separately. As before, the drum tracks were mixed at different levels with the accompaniment (all the remaining instruments and vocals). The onsets of the bass drum and of the snare drum were annotated, and MEME is expected to find the onsets and cluster both kind of events. Thus, this could be used to infer rhythmical patterns (since they are mostly determined by the pattern of bass and snare drums). Results (averaged among all mixes) are shown in Fig. 4 for all background levels and on the right side of Table 1 for 25 dB background level. For clarity, only the PCA results (which were consistently better) are shown in the figure. Again, PCA 3/1/m(15) is clearly the best configuration, reaching $F = 41.59\%$ at 25 dB and $F = 43.58\%$ at 40 dB. The overall performance is lower than in the environmental detection task. This is due to the fact that the mixes are more demanding: on the one hand, the background is considerably more complex, on the other, the bass and snare events in the drum tracks are often mixed with other percussive sounds such as hi-hats.

In all cases, the whole system is computationally efficient and performs all the processing stages (feature extraction, dictionary learning, sequence generation, motif discovery and index mapping) faster than real time. The ratios are of around 0.6 real time for the PCA 3/1 configuration and of around 0.7 real time for the PCA 3/1/m(15) configuration on a 3 GHz CPU with 8 GB RAM.

5. CONCLUSIONS AND OUTLOOK

We have proposed a new method for the unsupervised detection of recurrent sound events in a mixture, based on symbolic sequence representations and on a genetic motif discovery algorithm called MEME. Several dictionary learning and sequence representation methods have been evaluated, with the combination of PCA with metasequences as the best performing setup for all experimental cases. Conveying the inter-layer structural information via metasymbols significantly improves performance over using a single-layer sequence. The system has been tested in an environmental sound detection task and in drum onset detection.

Future work will focus on improving the robustness of the method against high noise levels, studying the stability of the algorithms (they are very sensitive to parametrization), and studying more closely the appropriateness of the feature space with respect to the used dictionary learning method. The fact that the performance curves are fairly oscillating suggests using larger evaluation databases in the future, to observe smoother trends.

6. REFERENCES

[1] P. D'haeseleer, "How does DNA sequence motif discovery work?," *Nature Biotechnology*, vol. 24, pp. 959–961, 2006.

[2] A. Ewert, S. Müller, and R.B. Dannenberg, "Towards reliable partial music alignments using multiple synchronization strategies," in *Proc. Int. Conf. on Adaptive Multimedia Retrieval (AMR)*, Madrid, Spain, 2009.

[3] J.J. Burred, "Genetic motif discovery applied to audio analysis," in *Proc. IEEE Int. Conf. on Audio, Speech and Signal Processing (ICASSP)*, Kyoto, Japan, March 2012.

[4] T.L. Bailey and C. Elkan, "Fitting a mixture model by expectation maximization to discover motifs in biopolymers," in *Proc. Int. Conf. on Intelligent Systems for Molecular Biology*, 1994.