# GENETIC MOTIF DISCOVERY APPLIED TO AUDIO ANALYSIS

*Juan José Burred*

Audionamix
114, avenue de Flandre, 75019 Paris, France
juan.jose.burred@audionamix.com

## ABSTRACT

Motif discovery algorithms are used in bioinformatics to find relevant patterns in genetic sequences. In this paper, the application of such methods to audio analysis is proposed. In the presented system, sounds are first transformed into a sequence of discrete states, corresponding to characteristic spectral shapes. The resulting sequences are then subjected to the MEME algorithm for motif discovery, which estimates a structured statistical model for each found motif. The system is evaluated in two tasks: the discovery of repetitive patterns in a large sound database, and the detection of specific audio events in an audio stream. Both tasks are unsupervised and demonstrate the viability of the approach.

***Index Terms***— Sequence motif, audio event detection, audio similarity, bioinformatics

## 1. INTRODUCTION

In bioinformatics, *motif discovery* is a set of techniques aimed at finding relevant patterns in genetic sequences [1]. Often, such sequence motifs guide important biochemical processes or largely determine protein structures. A genetic sequence is typically represented as a long string of letters out of an alphabet of 4 letters (for DNA sequences) or of 20 letters (for amino acid sequences). Motif discovery algorithms perform a statistical analysis of such strings, together with multiple comparisons between string segments, to derive a set of candidate motifs. A found motif is usually represented by a Position Weight Matrix (PWM) which contains the likelihood of appearance of each letter at each position in the motif.

In the present contribution, the application of genetic motif discovery to the analysis of sounds is proposed. Given a relevant representation of sound as a sequence of discrete states denoted by letters, the use of motif finding tools is attractive for a number of reasons. First, motif discovery algorithms account for the relative variability between instances of candidate motifs. Indeed, exact matches are rare in genetic sequences, and the same can be said about instances of realistic acoustic events. Also, some powerful algorithms are able to find motifs with few parameters, automatically determining parameters such as motif length, total number of motifs, and number of motif *sites* per sequence (in this context, a *site* is the position in a sequence where an instance of the motif starts). A further important advantage is that motif discovery algorithms are highly optimized, well established, and able to handle very large sequence databases, which are typical in bioinformatics.

Motif finding, while related, should not be confused with *sequence alignment*. The goal of the latter is to find the position of similar regions between two or more sequences, while motif finding derives an explicit statistical model describing the high-similarity regions, and can find interesting structures from scratch, without any initial sequence reference or query. Genetic sequence alignment has often been applied to audio (see, e.g. [2]). This is not the case for motif discovery, whose application to audio is novel.

This article presents an audio analysis system that first converts a database of sounds into a database of letter sequences, each letter representing a prototypical spectral shape, and then makes use of a genetic motif discovery algorithm to find short, repetitive sound events that are prominent in the database. A motif thus represents a characteristic temporal sequence of spectral shapes that is persistent among many different sound files. The system is then evaluated within two different contexts of use. It is first used as a tool for sound motif discovery in a large and heterogeneous sound effect database in a fully unsupervised way. This can be useful in an efficient database browsing scenario or for musicological or creative purposes. Secondly, the system is used for an audio event detection task, where the goal is to detect a set of specific short audio events within an audio stream. Previous methods aimed at such a task include fingerprinting [3] and Matching-Pursuit-derived features combined with Locality-Sensitive Hashing [4]. Further potential applications of this second scenario are click detection in old recordings and musical onset detection.

The motif finding method chosen for the proposed system is MEME (Multiple Expectation-Maximization for Motif Elicitation) [5, 6]. Apart from being one of the best established motif finding tools, the statistical model on which MEME is based was deemed adequate to detect sound events. Indeed, MEME uses a two-component mixture model, one for the motifs and one for the *background*, which in genetics corresponds to non-informative subsequences, and in sound to uninteresting segments between salient events.

Fig. 1 shows an overview of the audio motif discovery system. It is divided into two main parts. The *sequencing* part converts a sound database into a set of letter sequences. The *motif finding* part analyzes that set of sequences and outputs a database of sequence motifs with their corresponding audio segments.

## 2. SEQUENCING

The goal of the sequencing subsystem is to convert a set of sounds into a set of letter strings, each string corresponding to one sound file. In genetics, the consecutive nucleotides (in DNA sequences) or amino acids (in protein sequences) are represented by letters, and the position in the sequence denotes the position in the DNA molecule. In the audio counterpart proposed here, the letters represent distinct spectral shapes, and the position in the string corresponds to time. Thus, a motif is a characteristic temporal evolution of spectral shape

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

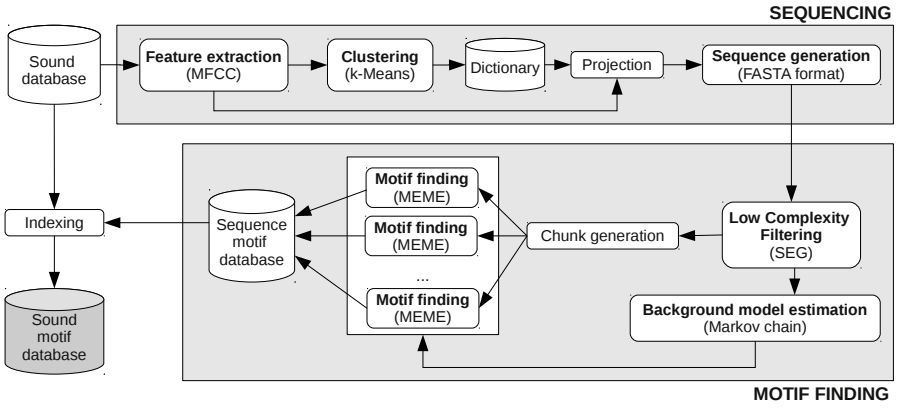



**Fig. 1**. Audio motif finding: system overview.

during a short interval. From this standpoint, motif finding can be considered a temporal modeling technique. Here, a *dictionary* is a small database containing representative spectral shapes, together with the mapping between them and the chosen letters. It should be noted that in contrast to bioinformatics, where letters are standardized and correspond to specific nucleotides or amino acids, here the assignment of letters to spectral shapes is arbitrary.

### 2.1. Dictionary learning

The dictionary is learned by performing a frame-wise feature extraction on the whole sound database and subsequently performing vector quantization on the features. The first 12 Mel Frequency Cepstral Coefficients (MFCC), without the first (energy) coefficient, were used as features. K-Means clustering was performed on the feature vectors. The centroids of the clusters are chosen as the dictionary elements. Thus, the number of clusters $K$ is also the number of dictionary elements and of the letter alphabet $A = \{a_1, a_2, \ldots, a_K\}$.

If the sound database is large, a downsampling of the feature vectors is needed prior to clustering to avoid computational overload. Thus, once the dictionary elements have been obtained, the original feature vectors have to be projected onto the dictionary vectors to obtain a similarity matrix between each feature vector and each dictionary element. To this aim, cosine similarity was used. In future versions of the system, it is planned to use more efficient clustering methods, such as mini-batch k-Means.

### 2.2. Sequence generation

Given the feature-to-dictionary similarity matrix, the letter corresponding to the closest dictionary element for each feature vector is chosen, obtaining a long string. The strings are exported into text files following the FASTA format convention, which is standard in bioinformatic software. Each file sequence is preceded by a header that can contain metadata describing the sounds.

## 3. MOTIF FINDING

### 3.1. Low Complexity Filtering

Prior to the actual motif finding, some preprocessing steps are needed. First, the ensemble of sequences are subjected to Low Complexity Filtering (LCF). This removes non-informative sections of the sequences, consisting mostly of long repetitions of the same letter, with few occasional changes to other letters. In an audio context, with MFCCs as features, long strings of the same letter correspond to a static spectral envelope, which is structurally uninteresting. Failing to remove those sections can mislead the motif finding algorithm to find one-letter motifs, since they are in fact repetitive, prevalent patterns. LCF is also crucial in bioinformatic motif finding [5]. A popular LCF algorithm from that area was used, called SEG [7].

### 3.2. The MEME algorithm

The basic assumption of the MEME algorithm[1] is that every subsequence of length $W$ in the database $X$ is generated from a statistical mixture model defined as the weighted sum of a motif model and a background (non-motif) model. Each substring is assumed to have been generated from either one of the models. The distribution for a given substring $\mathbf{x}_i \in X$ of length $W$ can thus be expressed as

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

where $\boldsymbol{\theta}_M$ is the parameter vector of the motif model, $\boldsymbol{\theta}_B$ is the parameter vector of the background model, $\lambda$ is the probability that the substring was generated by the motif model, and $\boldsymbol{\theta}$ is the global parameter vector $\boldsymbol{\theta} = \{\lambda, \boldsymbol{\theta}_M, \boldsymbol{\theta}_B\}$.

The motif model is described by a set of multinomial distributions, one for each position in the motif. In other words, each position $j$ in the motif, for $j = 1, \ldots, W$ is described by a multinomial distribution of parameters $\boldsymbol{\theta}_{M_j} = \{f_{j1}, f_{j2}, \ldots, f_{jK}\}$, where $f_{jk}$ is the probability of letter $k$ at position $j$. The multinomial distributions are assumed independent, giving a motif probability

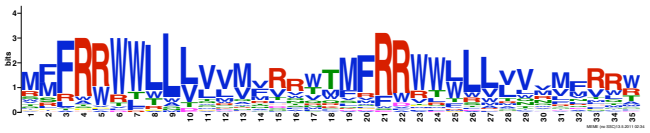
$$p(\mathbf{x}_i|\boldsymbol{\theta}_M) = \prod_{j=1}^{W}\prod_{k=1}^{K} f_{jk}^{I(k,x_{ij})}, \quad (2)$$

where $I(k, x_{ij})$ is the indicator function

$$I(k, x_{ij}) = \begin{cases} 1 & \text{if } x_{ij} = a_k \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

The background model is also a multinomial distribution, with the difference that its parameters do not depend on the position. Thus, it is described by a single parameter vector

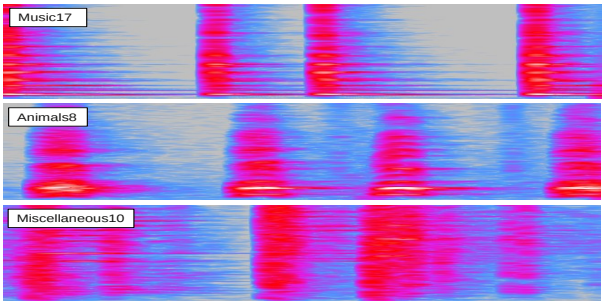
[1] Strictly speaking, MEME is the software implementing a motif finding algorithm called MM. Since MEME is the better-known acronym in bioinformatics, it will be used here to denote both algorithm and implementation.



(a) Sequence logo.

| Name | Start | *p*-value | Sites |
|---|---|---|---|
| Music17 | 696 | 1.57e-22 | FRRWTTLLLW MFFRRWWLLLLWMFRRWRMFRRRWWLLLVRMFRRW WFFFRWWLLL |
| Music17 | 619 | 2.63e-20 | LLLVMMRRWW MFFRRWWLLVVNMMRRRWMFFRRWWLLVVNMFRRW WMFFRRWWLL |
| Music17 | 506 | 2.63e-20 | FRRWTLLLVV MFFRRRWLLLVVMFRRWTMFRRWWLLLVNMFRRW WRMFRRWWLL |
| Construction7 | 991 | 9.49e-17 | LLVQVVTTFF RRFRRWWWLLTVVVVNTTMFRRWWWLLVVVNVNT NRFRRWWWWL |
| Construction7 | 422 | 9.49e-17 | LLVLTEETMM QMFRRWWWWLLLVVVVTTMFRRWWWLLLVVVQVQM FFRRWWWLLL |
| Music17 | 447 | 3.08e-15 | LLVWMMRWTT MFRRWTLLLVVMMRRWTTMFRRWLLLLVVRMFRRR WTLLLVRMRR |
| Footsteps19 | 1379 | 3.08e-15 | TAAFRQQMSS MFFAAWWRALTFMSFFWTQWRRAWTSAMFFFEALA AALRMSSMFA |
| Animals8 | 259 | 7.35e-14 | TMFMRWWLLV TMFRRWWLLLVLMFRRWWLVRFRRWLLTLTRMMRW WLLVVQVTVR |
| Animals8 | 221 | 7.35e-14 | TLLLVLQLLQ RFFRWWTTLLLVQTVLVWMFFRWWLLVVTMFMRWW LLVTMFRRWW |
| Footsteps19 | 1131 | 7.35e-14 | LLWRFRQWMF MFFALWWRLTVRHMSAQTLARRVQRSAMADEETWA ETMMMFATEW |
| Miscellaneous10 | 418 | 7.35e-14 | LLMSSMMSSF MSFRRRRWLLSSMRRRSSSFSRWWFFWTWWAARRT SSSFMMRSSS |

(b) Sequence motif sites.




(c) Spectrograms of audio motif sites (selection).

**Fig. 2**. Example of found motif in a large sound effects database, with a selection of corresponding sequence sites.

$\boldsymbol{\theta}_B = \{f_{01}, f_{02}, \ldots, f_{0K}\}$. Its probability is

$$p(\mathbf{x}_i|\boldsymbol{\theta}_B) = \prod_{j=1}^{W} \prod_{k=1}^{K} f_{0k}^{I(k,x_{ij})}. \quad (4)$$

Given the MEME model, motif finding is formulated as a Maximum Likelihood (ML) optimization problem via an Expectation-Maximization (EM) algorithm. To that aim, a set of latent variables $Z$ has to be defined in order to optimize the joint likelihood $p(X, Z|\boldsymbol{\theta})$. The latent variables are defined [6] as the set of binary indicators $z_{im}$ of the starting positions of the candidate motifs ($z_{im} = 1$ if a motif instance starts at position $m$ in the length-$M_i$ subsequence $\mathbf{x}_i$ and 0 otherwise). Thus, the ML problem is defined as the maximization of the log-likelihood

$$\begin{aligned} \mathcal{L}(\boldsymbol{\theta}|X, Z) &= \sum_{i=1}^{N} \sum_{m=1}^{M_i} \{z_{im} \log(p(\mathbf{x}_{im}|\boldsymbol{\theta}_M)\lambda) \\ &+ (1 - z_{im}) \log(p(\mathbf{x}_{im}|\boldsymbol{\theta}_B)(1 - \lambda))\}, \quad (5) \end{aligned}$$

where $N$ is the total number of subsequences in the database. See [6] for a detailed derivation from Eq.5 of the E and M steps.

Apart from the actual EM optimization, the MEME algorithm includes a number of heuristics to find good initialization values for the parameters, good motif starting positions, to automatically select the motif length $W$, which can be different among motifs, and to determine the number of motif instances per motif [6].

### 3.3. Background model estimation

As part of the overall parameter vector, the background model parameter $\boldsymbol{\theta}_B$ is optimized during EM. By default, it is initialized by measuring the letter frequencies in the input sequence database. In addition to the letter frequencies, MEME offers the possibility to use first-order letter transition probabilities (i.e., an Markov chain) for such initialization. These can be either estimated on the input sequence database, or estimated externally and passed as a vector.

The latter option is used here in order to cope with the following problem. The computational cost of MEME is high if the number of letters in a sequence exceeds a certain limit. Thus, the single FASTA file containing the whole database after LCF is cut into chunks, allowing running one instance of MEME per chunk. This makes computation tractable, but on the other hand it prevents MEME from observing the whole database when searching for motifs. Thus, to provide MEME with some prior information about the database as a whole, a Markov chain is estimated from the full database and passed to every instance of MEME, as indicated in the lower part of Fig. 1.

### 3.4. Sequence logos

For each found motif, MEME outputs the estimated model parameters $\boldsymbol{\theta}_M$ in the form of a PWM containing letter-position probabilities, together with a list of motif sites and a statistical significance value for each site (the *p-value*). A *sequence logo* is a graphical representation of a PWM. The columns of a sequence logo correspond to the position in the motif, and the height of the individual letter in each column is proportional to its probability of appearance at that position. In addition, the total height of each column denotes the information content $R_j$ in bits at that position, given by

$$R_j = \log_2 K + \sum_{k=1}^{K} (f_{jk} \log_2 f_{jk}). \quad (6)$$

An example of sequence logo is shown on Fig. 2(a). The colors used by MEME for display purposes are based on shared biochemical properties between the amino acids associated to the letters. In the audio application presented here, the colors are arbitrary and have no other purpose than to clarify letter transitions.

As the last step of the system, the motif site indices and lengths are mapped to the stored audio indices, and a set of sound files is generated by cutting the original sounds at the appropriate intervals.

## 4. APPLICATION TO AUDIO MOTIF DISCOVERY

The system was first used for the unsupervised search of motifs in a large sound database. The used database was the Sound Ideas Series 6000 General Sound Effects Library[2], comprising 3273 sound effects. For feature extraction, a Blackman analysis window of 40 ms with a hop size of 20 ms was used, and 12 MFCCs per frame were extracted. The alphabet was of size $K = 20$, which is the largest size allowed by MEME, corresponding to the 20 different amino acid symbols. The performance will likely benefit from larger alphabet sizes, allowing a more adequate representation of complex sounds. As part of future work, MEME will be modified to that aim.

[2] http://www.sound-ideas.com

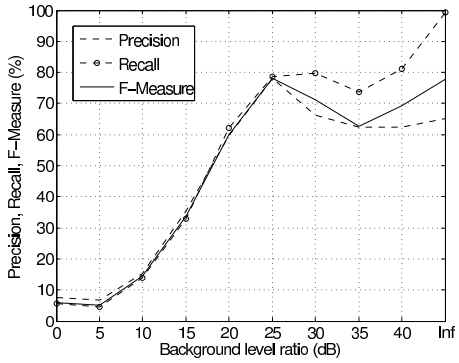



**Fig. 3**. Audio event detection results.

A collection of motifs and sound segments corresponding to motif sites was obtained. In many cases, the system detected very similar occurrences of a repetitive pattern inside the same sound effect file, or among very similar files. More interesting results from the point of view of search-by-similarity are, however, when sites of a given motif occur in sounds of very different classes. Fig. 2 shows an example of such a result, where a similar pattern, in terms of both rhythmical structure and timbre, was found in sound files such as a cowbell rhythm (marked as Music17 in the figure), a bird noise (marked as Animals8) and a sword fight (marked as Miscellaneous10). Fig. 2(a) shows the logo for this motif, and Fig. 2(b) shows a selection of motif sites, sorted by $p$-value (the lower $p$-value, the more statistically significant the site is). Fig. 2(c) shows the spectrograms corresponding to three of the found sites. The sound segments corresponding to this and other examples can be listened to in a companion website[3].

## 5. APPLICATION TO AUDIO EVENT DETECTION

As a second application, the system was used to detect similar, repetitive events in an audio file [3, 4], in an unsupervised way. To that end, an artificial 15 second sound scene was created, consisting of a background (ambiance sound in a park) mixed with occasional dog barks (8 in total) and 15 hammer strikes. The goal was twofold: to find the onsets of the bark and hammer events, and to cluster them into two classes. The events were manually annotated, and the experiment was repeated with different levels of the background ambiance to test the robustness of the system against mixed sounds. In this experimental setup, the "sound database" consists of only the input mixture sound. A 10 ms window length and 5 ms hop size were used. LCF was bypassed, as it was removing most of the letters with such a small database. The alphabet size was $K = 8$. In this task, the system is computationally efficient, and a full run takes roughly 1.3 times real-time on a 2.4 GHz Quad CPU with 8 GB RAM.

The performance was measured in terms of class-wise precision ($P$), recall ($R$) and $F$-measure of the detected onset positions compared to the annotated onsets within an error window of 40 ms. To compensate the randomness introduced by the algorithm initializations (notably k-Means), for each background level the experiments were repeated 10 times and the measures averaged. Fig. 3 shows the $P$, $R$ and $F$ values averaged among the two classes and among all experiment runs, as a function of the peak-to-peak amplitude ratio (in dB) between the events signal (barks plus hammer strikes) and the background signal. It can be seen that for high background levels (ratios from 0 to 20 dBs), precision and recall are balanced but always below 70%. For low background levels (higher dB ratios), recall always outweighs precision, indicating a high number of false positives. The rightmost point in the graph corresponds to an infinite level ratio (no background sound). The system is able to well detect and cluster the events down to a level ratio of 25 dB (the obtained measures for 25 dB are $P = 78.01\%$, $R = 78.67\%$, $F = 78.07\%$). For lower ratios, the background interferences are too high and disturb the motif finding process.

[3] `http://audionamix.com/AudioHelixMotifFinding1/`

## 6. CONCLUSIONS

Genetic motif discovery can be applied to audio analysis for a computationally efficient and unsupervised detection of repetitive sound events with temporal and spectral similarity. In the proposed framework, a motif can be considered a detailed statistical model of the temporal evolution of the spectral envelope. This is a novel application of genetic motif finding, and its usefulness has been demonstrated in two tasks: discovery of characteristic events in a large sound database *from scratch*, and onset detection of recurrent, salient audio events in a stream.

Once the validity of the approach has been proved, further, extensive optimizations are needed by collecting and annotating larger audio event databases, and by benchmarking results with alternative methods. Also, further research will be aimed at improving the robustness of the system against high background noise levels. To that aim, the sequence generation and statistical model will be adapted to better represent sound mixtures: the interleaved model used by MEME (alternating segments of motifs and background) could be reformulated as an additive model of different symbolic levels, corresponding to the mixed sounds.

## 7. ACKNOWLEDGMENT

The author would like to thank Torbjørn Rognes, Trevor Clancy and Håvard H. Hauge from Sencel Bioinformatics (Oslo, Norway) for the fruitful discussions.

## 8. 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. D. Dannenberg, "Towards reliable partial music alignments using multiple synchronization strategies," in *Proc. Int. Conf. on Adaptive Multimedia Retrieval (AMR)*, Madrid, Spain, 2009.

[3] J. Ogle and D. Ellis, "Fingerprinting to identify repeated sound events in long-duration personal audio recordings," in *Proc. IEEE ICASSP*, Honolulu, USA, 2007.

[4] C. Cotton and D. Ellis, "Finding similar acoustic events using matching pursuit and locality-sensitive hashing," in *Proc. IEEE WASPAA*, New Paltz, USA, October 2009.

[5] T. 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.

[6] ——, "The value of prior knowledge in discovering motifs with MEME," in *Proc. Int. Conf. on Intelligent Systems for Molecular Biology*, 1995, pp. 21–29.

[7] J. C. Wootton and S. Federhen, "Analysis of compositionally biased regions in sequence databases," *Methods Enzymol.*, vol. 266, pp. 554–571, 1996.