Variability Tolerant Audio Motif Discovery

Armando Muscariello, Guillaume Gravier, Frédéric Bimbot

Variability tolerant audio motif discovery

Armando Muscariello, Guillaume Gravier, Frédéric Bimbot

IRISA (CNRS & INRIA), Rennes, France
METISS, 'Speech and Audio Processing' research group

Abstract. Mining of repeating patterns is useful in inferring structure in streams and in multimedia indexing, as it allows to summarize even large archives by small sets of recurrent items. Techniques for their discovery are required to handle large data sets and tolerate a certain amount of variability among instances of the same underlying pattern (like spectral variability and temporal distortion). In this paper, early approaches and experiments are described for the retrieval of such variable patterns in audio, a task that we call audio motif discovery, for analogy with its counterpart in biology. The algorithm is based on a combination of ARGOS [4] to segment the data and organize the search of the motifs, and a novel technique based on segmental dynamic time warping to detect similarities in the audio data. Moreover, precision-recall measures are defined for evaluation purposes and preliminary experiments on the word discovery case are discussed.

Key words: audio pattern discovery, variable motif, dynamic time warping, normalized edit distance

1 Introduction

1.1 Motivation

Discovery of repeating patterns for multimedia indexing is an emerging research field. The increasing possibility to capture and store large amounts of multimedia documents has led to the adoption of strategies to quickly access, process and browse through massive data sets. Identification of patterns that structurally characterize a multimedia archive aims at coherently organizing the collection by representing the archive through a set of specific, recurrent items. Recent work on audio thumbnailing of music catalogs point towards this direction [1][2]. Moreover, in many cases, learning the structure of a process by pattern discovery can be very useful in seeking a model that reflects the properties of the source that has generated the process itself. This is roughly what is done in computational biology, where the extraction of meaningful patterns (usually referred as motifs) in massive amounts of DNA and protein sequences plays a key role in the analysis and understanding of important biological functionalities [5].
Allowing only identical patterns to be recognized would dramatically limit the potential applications of motif discovery. For example in comparative genomics, most of the time, patterns are allowed to present wild cards or indels, *e.g.* they

are not necessarily identical, but present a certain degree of variability.
Our interest is focused on the retrieval of such recurring patterns in audio streams or data sets, a task that, for analogy with its counterpart in genomic, we call audio motif discovery. Typical examples in streams are repetition of jingles, advertisements, or even entire shows broadcasted multiple times in a day, whose identification allows for customization of the stream (by skipping commercial, for example). Identification of words and verbal expressions that inherently characterize a news, a lecture, a movie, is useful for summarization and enable the user to fastly browse trough the audio archive without relying on a transcribed version of the data. Techniques for retrieval of such motifs are required scalability to manage large data sets, flexibility to handle the large variety of motifs lenght, and robustness to sources of variability that make it difficult to detect multiple copies of the same motif (temporal distortions, spectral variability of the human voice). We propose in this paper one of such techniques that has been preliminary tested on the task of word discovery in speech and we plan to verify its performance in different audio motif discovery experiments.

1.2 Related works

In the last few years, only few work have addressed and formalized the problem of unsupervised audio motif discovery.
In [3] an algorithm is proposed for motif discovery in time series, but the search is performed on a symbolic, intermediate level that is clearly a limitation for the application to the audio signal. In ARGOS [4], a scalable approach is used to detect repetition of multimedia objects in streams by only considering the audio portion, but variability is not taken into account as objects are supposed to be identical recordings broadcasted multiple times over the day.
In [6] fragments of speech are compared pairwise at the acoustic level by a segmental DTW (SDTW) and the output is used to build an adjacency graph (with times indexes as nodes and DTW scores as edges) followed by a clustering phase. However, this approach does not scale well for increasing large data sets as the number of comparisons grows quadratically with the number of segments. Moreover, SDTW shows higher complexity than conventional DTW.

1.3 Outline

The paper is organized as follows: a short introduction is done to specify the elementary subtasks that compose the problem, followed by the description of ARGOS, a procedure to segment the stream and organize the search. The main contribution of our work is in subsection 2.2, 2.3, 2.4, where three variants of a new segmental DTW algorithm are introduced for automatically discovering similarities in acoustic fragments. In section 3, a framework to evaluate the performance is proposed and preliminary results on a small data set are next presented. Finally, ideas for improvements and developments of the current work are discussed.

2 Description of the algorithm

The audio motif discovery can be conceptually organized in four different subtasks:

1. the segmentation of the data into smaller segments to be compared for similarity detection.
2. the transformation of the raw data in an alternative, less redundant representation suitable for the comparisons.
3. the definition of a similarity measure and the inference of a proper threshold to discriminate (dis)similarity, that is to detect instances of the same motif.
4. the search procedure, that is the structural way to organize the comparison of the segments.

We tackle all these aspect in this section.
First, we remark the notable difference between the motif discovery and the search for previously known patterns (query by content) in stream or data sets, a problem well addressed in the scientific literature. In motif discovery, the searched objects (the queries of the search) are not known a priori, but must be inferred from the stream itself in an unsupervised way. As motifs endpoints or even presence in the stream is unknown, naive exhaustive strategies imply candidate motifs of every possible length in every part of the stream to be assumed as queries and searched along the entire stream. This approach is clearly unfeasible even for small data sets.
Alternatively we resort on ARGOS, a general purpose strategy that exploit the intrinsic repetitiveness in streams to efficiently segment the data and organize the search.

2.1 The ARGOS approach

In ARGOS fixed length motif candidates (queries) are used, supposed to either coincide with the motif or include the motif as a portion of it. Moreover, the search for each query is not performed over the entire stream, but is rather restricted on its near future (or recent past) and a library is incrementally build where detected motifs are stored and used for retrieval of long term matches.
More specifically, at each step of the process, a portion of the incoming audio stream is broken into a pair query-buffer. The query is a segment of the portion of stream under processing, which is supposed to completely contain the motif. The buffer is the audio portion adjacent to the query and it represents the search-space where the query is seeked into.
The underlying assumption is that meaningful patterns repeat frequently, at least in a part of the stream, thus they are likely to repeat in their near future [1].
When a repetition of the current query in the buffer is detected, a reference model of the common pattern (the found motif) is stored in a library and used

[1] or in the recent past, as in the original work

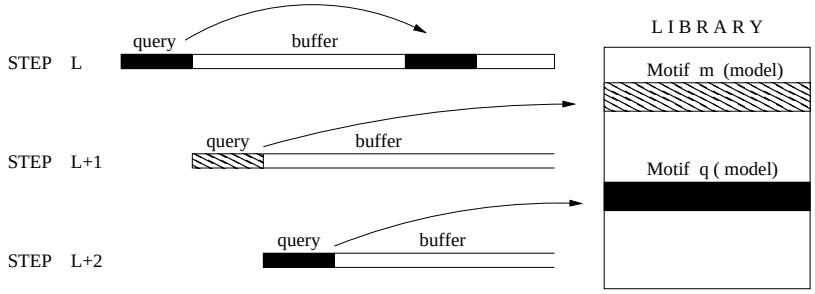

Fig. 1. ARGOS framework: at step L, the current query is not detected in the library but it is found in the buffer and the corresponding reference model is stored in the library as the q-th motif. At step L+1 query and buffer shift along the stream of a query length and the current query is retrieved in the library by comparison with the m-th motif. At step L+2 the new query, which happens to be another occurrence of the q-th motif, is retrieved by direct search in the library.

for future comparisons. As the process evolves, query and buffer shift along the audio stream, and the new query is first seeked into the library (by comparison with the stored models) and, if not found in the library, in the search buffer (as illustrated in figure 1). It follows that each query is searched, at most, in the K motifs currently stored in the library plus the search buffer, unlike the exhaustive approach that implies a number of comparisons that quadratically grwos with the number of queries.
In the original work, the search is performed on the audio portion by time correlating distorted versions of the speech signal, obtained by only retaining a small part of the audio spectrum (about 200 Hz centered around the sixth Bark band). Such a reduction technique can decently perform only in a context where a very few samples are needed to discriminate (dis)similarity, that is, occurrences of the same motif are supposed to be practically identical and completely different from other motifs; it is therefore unsuitable in a word discovery task, and, in general, in a scenario where instances of the same motifs can exhibit a certain amount of variability. For this purpose, we propose to resort to a more accurate spectral representation of the audio signal (MFCC) and exploite the potential of dynamic programming for pattern identification. Several implementations of a new segmental DTW technique are carefully described in the following.

2.2 Segmental locally normalized DTW

DTW is a widely used technique for pattern recognition. In the classical version, it is used to detect similarity between two motif templates, a and b, by computing spectral frame vectors $\{u_i\}_{i=1}^{M}$ and $\{v_j\}_{j=1}^{N}$, and the frame-to-frame distance matrix $d(i,j), 1 \leq i \leq M, 1 \leq j \leq N$. Applying recursively dynamic programming (DP) relations, a path $P = [(1,1), \cdots, (M,N)]$ of length $L(P)$ is found and the corresponding average weight $W(P) = (d(1,1) + \cdots + d(M,N))/L(P)$

is compared against a spectal threshold ϕ to decide if a and b are similar.
As in our framework we can not rely on an *a priori* segmentation of the stream into exact motifs, this approach is not suitable since it only works when motifs endpoints are well defined. Indeed, motifs boundaries are not known in advance. Our goal is to be able to find a path P from (i_s, j_s) to (i_e, j_e), with $1 \leq i_s \leq i_e \leq M, 1 \leq j_s \leq j_e \leq N$, relaxing the boundary constraint of the classical approaches that force starting and ending point to be respectively at $(1,1)$ and (M,N). We first consider the case $i_s = 1, i_e = M$, that is, we search a repetition of the whole vector u (query) into v (buffer).
The solution we propose relies on a heuristic that consists in locally minimizing the average weight of each path, both when selecting new starting points and when computing the paths itself, with same complexity as the conventional boundary-constrained approaches. We call it segmental locally normalized DTW (SLNDTW), as it allows for multiple paths with different starting points (Segmental) and it is based on a Local Normalization principle.
We define $L(i,j)$ as the length of the path starting from some $(1, j_s)$ up to (i,j), $D(i,j)$ the corresponding accumulated distance and $W(i,j) = D(i,j)/L(i,j)$ its average weight.
As a potential match can occur anywhere in v, we need a strategy to allow $j_s \neq 1, j_e \neq N$.We identify a starting point by comparing each cell $(1,j)$ with its left neighbour $(1, j-1)$ (which has been evaluated previously, as computation proceeds from left to right, as in classical DTW): if $d(1,j)$ is less than $W(1,j)$ (the weight of the path obtained by adding $d(1,j)$ to $D(1,j-1)$), then it is decided to start a new path from $(1,j)$ as a starting point of a potential matching sequence. Formally:
$\forall j, 1 \leq j \leq N$,

$$\begin{cases} D(1,j) = d(1,j) \\ L(1,j) = 1 \end{cases} \text{, if } d(1,j) < W(1,j)$$
$$\begin{cases} D(1,j) = D(1,j-1) + d(1,j) \\ L(1,j) = L(1,j-1) + 1 \end{cases} \text{, otherwise} \tag{1}$$

Except for $i = 1$, each path is computed by iteratively applying the DP relations following the local normalization paradigm, which consists in minimizing, at each point (i,j) of the computational grid $[1, \cdots, M] \times [1, \cdots, N]$, the weight $W(i,j)$, that is the quotient between the accumulated distance $D(i,j)$ and the path length $L(i,j)$. Formally:

$$W(i,j) = \min \left[\frac{d(i,j) + D(i-1,j)}{L(i-1,j)+1}, \frac{d(i,j) + D(i-1,j-1)}{L(i-1,j-1)+1}, \frac{d(i,j) + D(i,j-1)}{L(i,j-1)+1} \right] \tag{2}$$

The ending point (M, j_e) of a match is such that $W(M, j_e) < \phi, 1 \leq j_e \leq N$, where ϕ is a spectral threshold. If several such points exist, that is multiple occurrences of the query in the buffer occur, we just retain the first one and initialize a cluster in the library, modeling the motif as the average of the spectral frames put in correspondence by the DTW mapping. The other occurrences will

be detected later when assumed as queries and searched in the library, and the reference model will be updated as well by averaging with the newly detected instances of the motif.
Like in conventional DTW we only need to scan the distance matrix d once to compute D and L (W is D/L), differently from SDTW, where after paths computation, each diagonal band needs to be re-evaluated for subpath identification. Moreover, while in SDTW starting points are a priori selected by regularly sampling the first row of $[1, \cdots, M] \times [1, \cdots, N]$, in SLNDTW each cell $(1, j)$ is a starting point candidate.
It is worth noting that paths are not forced to be confined in diagonals of a pre-defined slope, and various local constraints can be applied, depending on the application, allowing for matches with different slopes.

2.3 Band relaxed SLNDTW

SLNDTW aims at finding matches of the query in the search buffer. In our framework, this approach would be effective only if motifs were of fixed length and coincide exactly with the query. This is a strong assumption and far from realistic applicative contexts. If the length assumption restricts the number of retrievable motifs, the mismatch in time synchronization between motif and query dramatically decrease performance, as increasingly high path weights result from even sligth timeshiftings. We propose here a modification of SLNDTW, band relaxed SLNDTW, that relaxes the boundary constraints of SLNDTW selecting starting and ending points in a group of rows (band), instead of a single one, thus allowing to retrieve motif with different lengths, as illustrated in figure 2.
This is achieved by dividing the grid $[1, \cdots, M] \times [1, \cdots, N]$ in three horizontal bands and selecting starting point in the first one and ending point in third, constraining all the paths to cross the second one. The starting band includes all $(i, j)|i \in [1, L_s]$, the central band includes points $(i, j)|i \in]L_s, L_s + L_c]$ and the ending band includes all points $(i, j)|i \in]L_s + L_c, M]$. Accordingly, motif lengths are allowed to vary from L_c to M.
More specifically:

1. $\forall (i, j)|i \in [1, L_s]$:

   if $d(i, j) < \left[\frac{d(i,j) + D(i-1,j)}{L(i-1,j)+1}, \frac{d(i,j) + D(i-1,j-1)}{L(i-1,j-1)+1}, \frac{d(i,j) + D(i,j-1)}{L(i,j-1)+1} \right]$

   then (i, j) is the starting point of a new path, otherwise it is added to the path that minimizes $W(i, j)$. Note that this a generalization of eq. (1), as the same condition is expressed by considering the whole neighbourhood of (i, j) rather than the single cell at its left $(i, j - 1)$.
2. $\forall (i, j)|i \in]L_s, M]$ compute path as in eq. (2).
3. $\forall (i, j)|i \in [L_c + L_s, M]$ select the ending point of a match, if any, as in SLNDTW, and reconstruct the corresponding path.

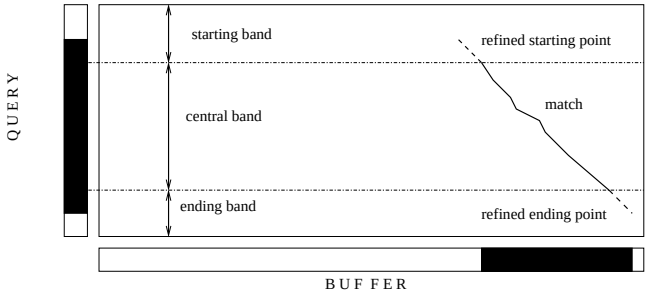

Fig. 2. Band relaexd SLNDTW: the motif completely includes the central band. After path reconstruction, boundaries are refined in the starting and ending band (dashed lines).

In addition, we have applied an heuristic for the refinement of the boundaries that consists in extending the found match by adding new frames at the boundaries (following the local normalization paradigm), as long as the average weight of the extended path does not increase too much.
Formally:

1. Consider the path P with $W(P) = W_o$ ending in (i_e, j_e).
2. Select in the neighbourhood of (i_e, j_e) (composed of $(i_e + 1, j_e + 1), (i_e + 1, j_e), (i_e, j_e + 1)$) the point that, added to P, minimizes $W(P)$, and add it to P as its new ending point.
3. If $W(P) < W_o + 10\% W_o$, then repeat the procedure from 1, otherwise remove the new ending point from P and stop the procedure.

The same approach applies when extending the path backward from its starting point (i_s, j_s).

2.4 Fragmental SLNDTW

Band relaxed SNLDTW does not constrain motif and query to coincide, but it still assumes the motif to be located in the middle part of the query, such that it completely includes the central band. A simple generalization of the previous versions of the algorithm, that we call fragmental SLNDTW, allows to retrieve the sought motif regardless of its position in the query, by first retrieving a portion of it, *e.g.* a fragment. SLNDTW detects a match whenever a query coincide with a motif. By using queries small enough to be included in the motif, then there exists at least one fragment of the motif that coincide with one of the queries and that can be discovered by SLNDTW. Indeed, if $L_{min} \leq L_{motif} \leq L_{max}$, partitioning a L_{max} long query in $L_{min}/2$ long subqueries ensures that at least a $L_{min}/2$ long fragment of the motif coincide with one of the subqueries, and it is therefore retrievable by conventional SNLDTW. The entire match can be recovered afterwards, by extending the corresponding path as in the boundary

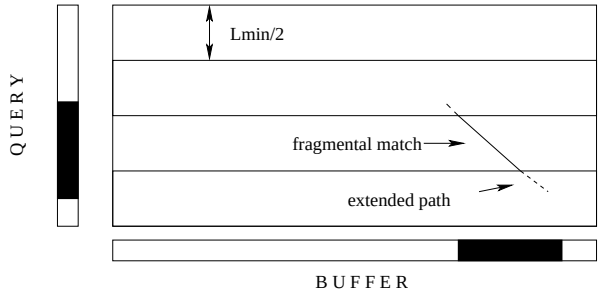

Fig. 3. Fragmental SLNDTW: partitiong the query in $L_{min}/2$ long subqueries ensures that a least a fragment of the motif coincide with a subquery. The entire match can be then recovered by extending the fragmental match.

refinement stage in Band relaxed SLNDTW.
For the sake of clarity we explicit the steps of the procedure and illustrate the scenario in figure 3:

1. divide the grid $[1, \cdots, M] \times [1, \cdots, N]$ in horizontal bands of length $L_{min}/2$, such as the i-th band includes all point $(i, j)|(i-1) \cdot L_{min}/2 + 1 \leq i \leq i \cdot L_{min}/2$.
2. perform a conventional SNLDTW in each band and reconstruct the found match, if any.
3. extend the path corresponding to the found match with the same heuristic used to refine boundaries in band SLNDTW.

This implementation of the technique has the advantage to enable the retrieval of a match whichever its position in the considered query, hence it shows higher flexibility than the two previous versions. It only constraints the motif minimum and maximum lenght, which is not a very limiting assumption in many applications.
Therefore, using an accurate spectral representation of the audio signal (MFCC) and combining the described method with ARGOS segmentation-search strategy, the motif discovery can be finally performed.

3 Evaluation

The evaluation of the performance relies on the analysis of the library of motifs constructed by the algorithm. We propose here a framework for the computation of a recall-precision curve. Precision aims at quantifying the level of purity of each cluster in the library, that is the ability of the algorithm to limit false hits as much as possible, while recall aims at measuring the ability to limit missed detection of motif's instances, or, equivalently, to retrieve, for each motif, as many exemplars as possible. In our framework, evaluation has been performed at the phonetic level relying on a transcribed version of speech data; accordingly

we have resorted to *normalized edit distance* d [7] and a phonetic threshold θ to verify the (dis)similarity between motifs found by the algorithm at the spectral level.
We introduce the following notation:

- LB_i: i-th motif of the library LB.
- $LB_{i,j}$: j-th instance of the motif LB_i.
- m_i: cardinality of LB_i.
- $d(LB_{i,j}, LB_{i,k})$: distance between $LB_{i,j}$ and $LB_{i,k}$.
- c_i: centroid of LB_i

The centroid c_i of LB_i is defined as:

$$c_i = LB_{i,p} \text{ where } p = \arg \min_{1 \leq j \leq m_i} \sum_{k=1}^{m_i} d^2(LB_{i,j}, LB_{i,k}) \tag{3}$$

The precision of the i-th motif is thus computed as:

$$P_i(\theta) = \frac{\left(\sum_j \delta \left(d(LB_{i,j}, LB_{i,p} < \theta \right) \right)}{m_i} = \frac{m'_i}{m_i} \tag{4}$$

where $\delta = 1$ if its argument is true, and 0 otherwise. It represents the fraction of instances $LB_{i,j}$ included in a sphere of center c_i and radius θ. The global precision $P(\theta)$ is the average of P_i over all motifs LB_i.
Let m''_i be the number of entities M over the entire phonetic transcription such as $d\left(M, LB_{i,j}\right) < \theta$. The recall of the i-th cluster is the ratio:

$$R_i(\theta) = \frac{m'_i}{m''_i} \tag{5}$$

and the global recall $R(\theta)$ is computed by averaging over all motifs of the library.

4 Preliminary experiments

The test data is composed of a 20 minute long French broadcast recording, sampled at 16 KHz. Words are uttered from different speakers (the conductor and the authors of the live reports) and no preliminary segmentation or pre-processing (like silence deletion) is performed. 13-dimensional MFCCs vector are extracted every 10 ms.
As the dimension of the processed file is quite small, even motifs occurring as few as 2 times are retained and considered for recall-precision evaluation: therefore the resulting numbers are not meant as statistically relevant measurements. Nonetheless, they are useful to evaluate and compare the performance of the different implementations of the algorithm.
In a 20 minutes bulletin, the time duration of each report is around 1 minute; as patterns inherently characterizing a 1 minute news have been supposed to

repeat in a few seconds, we have arbitrarily set the buffer length to 13 seconds for all runs of the algorithm. As queries and motifs are supposed to coincide in conventional SLNDTW, the query length has been set to an average word length (0.6 second). In Band relaxed SLNDTW, the motif is supposed to be located in the middle of the query. We have therefore used 1 second long queries. In the Fragmental SLNDTW, motifs are allowed to be located anywhere in the query. Given the flexibility of this last method, among all the possible choices, we have arbitrarily chosen 2 seconds long queries. Finally, in the Band relaxed version we have set $L_c = 0.3$ s and in the Fragmental SLNDTW we have set $L_{min}/2 = 0.3$ s.

The different versions of the algorithm have been tested for increasing values of the spectral thresholds ϕ, from $\phi = 8$ to $\phi = 12$. In some experiments we have noted that, even in the correct detection of two occurences of the same motif, certain phonemes at the boundaries of the two exemplars are not detected. In order to take into account the issue, we have empirically set the phonetic threshold θ to 0.35.

We have noted that, at least for $\phi \leq 10$, the resulting library of motifs exhibits a significative level of purity, as acoustically similar fragments are well grouped together. Motif length has mostly ranged from 0.45 s (set as minimum acceptable length)to 0.9 s. Example of retrieved motifs are: single words, usually including some phonemes from the preceding and following words (*les ambassadeurs du G*), or subwords shared in common by different words (*la position - la discussion*), or even small multi words locutions (*face a l'interdiction*), while the small breathings in between words have been notably the most frequenly retrieved pattern. In some cases we have noted different clusters representing the same underlying motif that the algorithm has failed to merge together, in particular for words uttered by different speakers. In several occasions, repeating words have not been detected at all, either because the size of the buffer has revealed to be too short to detect them, or because different exemplars of the same word have shown higher spectral distances than expected.

From a quantitative point of view, the results of the experiments are summarized in figure 4 and 5. In figure 4 the number of found motifs for a certain value of ϕ is shown for the different versions of the algorithm. As expected, this number is always higher for Band relaxed and Fragmental SLNDTW with respect to conventional SNLDTW, since in this last case no variability in motif length is allowed and consequently, only 0.6 seconds repeating words (the query length used) can be detected; moreover only perfectly aligned exemplars are likely to be found, as slight misalignments, as already noted, infer significative distortions.

In figure 5 it can be observed that, for increasing values of ϕ, the precision P decreases, as false hits appear more frequently. Even if the the experiment was conducted on a small data set, it is noteworthy the high value of purity suggested by the computed precision, in particular for $\phi < 11$ and for the last two versions of SLNDTW.

The behaviour of the recall parameter R, for different values of ϕ is less straightforward to understand.

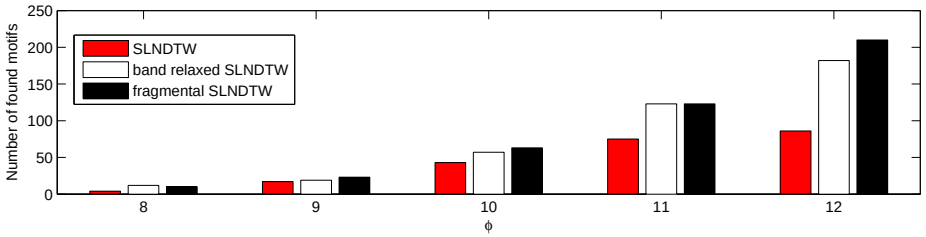

Fig. 4. Number of found motifs for the different algorithms and different values of θ

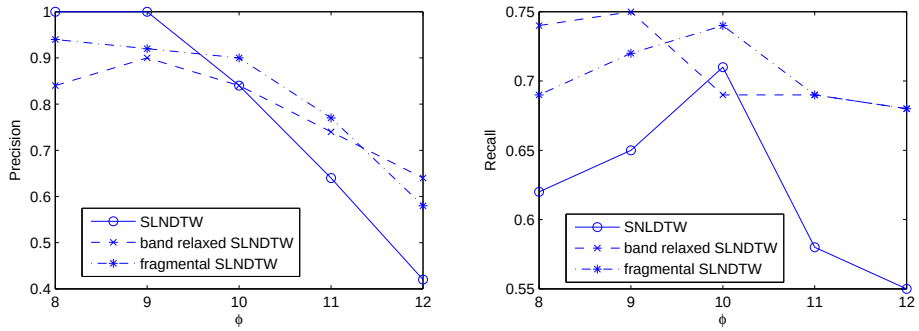

Fig. 5. Precision and recall curve for the different algorithms and different values of θ

Indeed, in SLNDTW and Fragmental SLNDTW this value tends to increase from $\phi = 8$ to $\phi = 10$ and then to fall for larger values of ϕ. The same transition can be observed between $\phi = 9$ and $\phi = 10$ for Band relaxed SLNDTW.
We initially predicted that an increase of ϕ would be followed by a substantial improvement of R at the expense of P, as more instances of the same underlying motif -as well as more false hits- are likely to be detected for higher values of spectral threshold; instead, the way the reference model is built and updated in the library strictly relates recall and performance measures, as averaging false hits with the reference model tends to progressively reduce its representativeness, leading to missed detection of true instances of the same motif. In synthesis, updating the reference model is highly prone to error propagation, when increasing the spectral threshold. Moreover, as defined as in eq. (5), the recall is only computed over the found motifs, not taking into accounts those motifs that algorithm does not detect at all, that should contribute each with a single recall $R_i = 0$.

5 Conclusions and future works

In this work, we have addressed and formalized the task of audio motif discovery. We have proposed an algorithm that combines ARGOS and three different

implementations of a novel DTW approach for audio similarities detection that seamlessly integrates into ARGOS. The algorithm has been tested on the word discovery case, and has shown promising results, at least in terms of precision. It exhibits a certain robustness to the typical spectral variability in speech when detecting multiple realization of the same word. We plan to investigate its performance in different audio motif contexts.
Large scale experiments are needed to validate the preliminary results here presented and to test the sensitivity of the algorithm to variations of main parameters. As far as the improvement of the current algorithm we note here that the most remarkable limitation of the current method is that it limits the pattern discovery problem to the search and identification of low distortion regions in the local distance matrix. However, we have noted that, for a variety of reasons (different speakers, environmental conditions and so on), same words at different points in the audio file present different values of (locally normalized) distance when compared against each other; that makes it difficult to set a fixed reliable spectral threshold to discriminate between false and true matches. However, we have discovered visual similarities in their local distance matrices, which are consistent in the majority of the compared instances, regardless of the distortion of the main diagonal. We plan to investigate the nature of these patterns to improve the recognition task, by exploiting the large corpus of techniques in the image processing literature. The ultimate goal is to build an adaptive model where different spectral thresholds are set for each motif in the library and updated as new instances are found. Moreover, in order to speed up the computation, technique for fast access to the library can be applied (for example, by storing the motif in order of decreasing frequency of occurrence), together with techniques for fast approximation of DTW.

References

1. Dannenberg, R.B., Hu, N.: Pattern Discovery Techniques in Music Audio. Third International Conference on Music Information Retrieval, pp.63-70 (2002).
2. Peeters, G: Deriving Musical Structures From Signal Analysis for Music Audio Summary Generation: Sequence and State Approach. Content Based Multimedia Indexing, (2003).
3. Lin. J, Keogh, E., Lonardi, S., Pratel, P.: Finding Motifs in Time Series. ACM SIGKDD (2002).
4. Herley, C.: ARGOS: Automatically Extracting Repeating Objects from Multimedia Streams. IEEE Transactions on Multimedia, VOL. 8 (2006).
5. Brazma, A., Jonassen, I., Eidhammer, I., Gilbert, I. Approaches to Automatic Discovery of Patterns in Biosequences, J. Comp. Biology, vol. 5, no. 2, pp. 279-305 (1998).
6. Park, A., Glass, J.R.: Unsupervised pattern discovery in speech: IEEE Transaction on Acoustic, Speech and Language Processing, VOL. 16, (2008).
7. Vidal, E., Marzal, A., Aibar P.: Fast Computation of Normalized Edit Distances. IEEE Transaction on Pattern Analysis and Machine Intelligence, VOL 17, NO. 9, (1995).