Measuring Structural Similarity in Music

Juan P. Bello, *Member, IEEE*

***Abstract*—This paper presents a novel method for measuring the structural similarity between music recordings. It uses recurrence plot analysis to characterize patterns of repetition in the feature sequence, and the normalized compression distance, a practical approximation of the joint Kolmogorov complexity, to measure the pairwise similarity between the plots. By measuring the distance between intermediate representations of signal structure, the proposed method departs from common approaches to music structure analysis which assume a block-based model of music, and thus concentrate on segmenting and clustering sections. The approach ensures that global structure is consistently and robustly characterized in the presence of tempo, instrumentation, and key changes, while the used metric provides a simple to compute, versatile and robust alternative to common approaches in music similarity research. Finally, experimental results demonstrate success at characterizing similarity, while contributing an optimal parameterization of the proposed approach.**

***Index Terms*—Audio signal processing, computer audition, music information retrieval (MIR), music structure analysis, sound similarity.**

I. Introduction

LARGE music collections are now as likely to be found in personal computers, as they are on the servers of distributors, digital libraries, and radio stations. Yet, despite the advent of new technologies and the vast quantities of music recordings now in circulation, the organization and retrieval of music is mostly driven by metadata, tags, and other textual descriptions that are often vague, inaccurate, and always insufficient to describe the complexities of music. In this context, audio-based analysis and retrieval can complement existing text-based systems, and help redefine how users search and interact with digital music content.

The field of music information retrieval (MIR) aims at extending the understanding and usefulness of music data, through the research, development, and application of computational approaches and tools. An important focus of research is measuring the similarity between music recordings, and its application to a wide variety of tasks such as query by example, playlist generation, auto-tagging, music recommendation, the organization and visualization of music collections, etc. Most content-based similarity models are based on the so-called "bag of features" approach, where sounds are described in terms of the general behavioral pattern of low-level signal features—e.g., spectral centroid, zero-crossing rate, mel-frequency cepstral coefficients (MFCC), RMS, etc.—regardless of the values they assume, or their temporal ordering and structure. This low-specificity approach has met with great success in the characterization of sound (or texture) similarity [1]–[4].

However, music audio is a highly structured information medium, containing sounds organized both synchronously and sequentially according to attributes such as pitch, timing, or timbre [5]. Sound configurations in music define notions of harmony, melody, style, and form, and can elicit emotional responses from listeners [6]. Therefore, the analysis and identification of the patterns and relationships that govern temporal organizations in music is not only essential to its retrieval, but also to understanding its composition, categorization, and impact.

This paper proposes a computational approach to measuring the similarity between musical recordings based on their global temporal structure. It consists of a signal processing stage, whereby a set of features are extracted from the audio signal; a representation stage, where recurrence plot analysis is used to identify patterns of repetition in the feature sequence; and a similarity stage, where a practical realization of the information distance between objects is used to compute the pairwise similarity between music recordings.

While structure analysis has been extensively researched in music information retrieval, this work is different in that it does not intend to explicitly segment the audio into sections. Instead, it seeks to measure similarity directly on an intermediate representation of the signal's structure. Its main novelty is in the combined use of delay-coordinate embedding, recurrence plot analysis and the normalized compression distance (NCD) for music analysis in general, and the task of structure-based retrieval in particular. Recurrence plots can be seen as extensions to the self-similarity matrices widely used in MIR, while NCD is presented as an alternative to previous metrics of music similarity. Finally, this paper contributes a systematic evaluation of feature extraction, embedding and recurrence analysis strategies, on a music collection of more than 3000 tracks, making it the largest study of structure-based music information retrieval to date.

The reminder of this paper is organized as follows. Section II discusses previous relevant work in music structure analysis and similarity, recurrence plots, and the normalized compression distance. Section III introduces the proposed approach and discusses the details of its implementation. Section IV describes the data and methodology used in our experiments, while their results are presented and discussed in Section V. Concluding remarks are given in Section VI. This work significantly extends both the method and evaluation previously published in [7].

Manuscript received July 12, 2010; revised November 19, 2010; accepted January 04, 2011. Date of publication February 10, 2011; date of current version July 20, 2011. This material is based upon work supported by the National Science Foundation under Grant IIS-0844654 and by the U.S. Institute of Museum and Library Services under Grant LG-06-08-0073-08. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Bryan Pardo.

The author is with the Music and Audio Research Laboratory (MARL), New York University, New York, NY 10012 USA (e-mail: jpbello@nyu.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TASL.2011.2108287

II. Previous Work

A. Music Structure Analysis

Most research on the automatic analysis of music structure departs from two main premises: that music is highly repetitive and that its structure can be described in terms of a concatenation of high-level segments such as phrases, sections, and movements. These assumptions effectively reduce the issue of structure discovery, whether for summarization, chorus finding or structure extraction, into one of segmenting and clustering repetitive patterns in the feature sequence. Strategies in the literature include the use of agglomerative clustering [8]; hidden Markov models (HMMs) combined with simple aggregation [9], string matching [10], k-means clustering [11] and Bayesian clustering of state histograms [12]; and more recently the use of sparse, convolutive non-negative matrix factorization [13].

Since its introduction to MIR in [14], one of the most popular strategies in music structure analysis is based on the analysis of self-similarity matrices. For an N-long data series, they are the $N \times N$ array of pairwise distances between its elements. In audio, these series can be time–frequency representations or sequences of feature vectors extracted from the signal. In these matrices, repetitions are often characterized by small distance values that tend to group into diagonal and block patterns. Beyond visualization, this property has been exploited for tasks as diverse as rhythmic analysis [15], automatic summarization [16], chorus detection [17], synchronization [18], and long-term segmentation [19].

B. Structure-Based Similarity

The literature, however, is notably scant regarding the use of structure for music retrieval, even more so for the use of self-similarity matrices as anything other than intermediate steps toward segmentation. An early exception is the work in [20], where the pairwise structural similarity between recordings is computed as the cost of aligning feature sequences using dynamic programming (DP). The approach is tested on 82 tracks of orchestral and piano music, resulting on best performance for long queries represented using their log-magnitude spectra.

In [21], this idea is extended by computing the DP-based distance between self-similarity matrices, which has the advantage of introducing a certain degree of key-invariance into the process, as changes of key have little effect on the relative distance between feature vectors. Since standard DP-alignment cannot be used for the comparison of self-similarity matrices, the authors propose a modified algorithm that aligns only across the main diagonal and accounts for the dependencies between rows and columns in the matrix. Results, on a database of ten MIDI-generated tracks, indicate robustness against key and tempo changes. This method was also tested in [22] for the task of retrieving a piece of music based on a symbolic description of its long-term structure. The approach is compared against another 2-D DP-based method and a modified Euclidean distance. Results, on a 260-strong dataset, are mixed, with good performance depending on prior knowledge of section boundaries in the query.

C. Recurrence Plots

Self-similarity matrices (SSMs), also known as distance plots, are closely related to the recurrence plots introduced in [23] for nonlinear time series analysis. Generally speaking, recurrence plots (RP) differ from self-similarity matrices in that they are not computed directly from the time series but from its embedding into a delay coordinate space, and in that they are usually binary. It should be clarified that binarization is a common postprocessing step in the analysis of SSM. However, we emphasize this difference since previous work in structure-based similarity does not use it. The computation of recurrence plots will be described in more detail in Section III-B. See [24] for an extensive review.

There are only a few previous applications of nonlinear time series analysis to music information. Examples include the use of delay coordinates for the visualization of musical sounds [25], timbre analysis and synthesis [26], [27], and the analysis of expressive timing in piano performances [28]. However, to the best of our knowledge, the work in [29] is the only previous example of explicitly using recurrence plot theory for music information retrieval. In this paper, the authors successfully apply cross-recurrence quantification analysis to the task of cover song identification, using a variant of RP aimed at pairwise comparison of time series. Their use of recurrence plot theory, and their methodical testing of system variables, are important inspirations for this work. However, our goal of modeling structural similarity instead of cover song identification means that we choose to use classic instead of cross-recurrence plots, as they are more adept at characterizing the structure of the time series; and we do not choose to quantify recurrence, but the amount of shared information between plot pairs.

D. Normalized Compression Distance

Comparing sequences according to their structure is a topic of interest in many disciplines, notably bioinformatics, where similarity between protein structures can be indicative of common origin or functionality. As in music, protein structures are generally characterized using a variant of recurrence plots known as *contact maps*, similar in all aspects but the lack of delay coordinate embedding [30]. The problem of comparing protein topologies using contact maps is known as *maximum contact map overlap*, with many proposed solutions in the literature (e.g., [31] and [32]). This paper adapts to music the one proposed in [33], which successfully uses the *normalized compression distance* (NCD) to measure similarity between contact maps. The NCD is discussed in detail in Section III-C.

Previous uses of the NCD for music analysis include the clustering and classification of symbolic (MIDI) data based on genre, style, and melody [34], [35], audio-based sound and music classification [36] and, with limited success, cover-song identification [37].

The present work is the first to combine the use of classical recurrence plots and the normalized compression distance for music information retrieval and to present NCD as a simple to compute and robust alternative to the DP-based methods that dominate previous approaches to structural similarity.

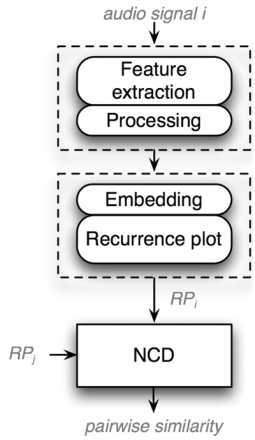

Fig. 1. Overview of the proposed approach.

III. Approach

Fig. 1 depicts the proposed approach, consisting of three main stages: first, features are extracted from the audio signal and processed to maximize performance; second, we characterize recurrences in the feature sequence via time-delay embedding and recurrence plot analysis; and finally, we measure the pair-wise similarity between these recurrence plots using the normalized compression distance. The details are explained in the following.

A. Feature Extraction and Processing

Most western music is based on the *tonality* system, which arranges sounds according to pitch relationships into spatio-temporal structures, such as melodies, chords, and keys [5]. The set of possible pitch values can be defined in terms of the attributes of height and chroma. Height characterizes the perceived pitch increases concomitant with increases in the frequency of a sound, while chroma relates to the cyclical perception of pitch as it moves from one octave to the other. According to this model, sounds whose frequencies are separated by an integral number of octaves occupy the same position, or pitch class, in the chroma circle [38].

Starting with [39], a number of techniques have been proposed for the characterization of the audio signal's energy content across pitch classes. The resulting representation, interchangeably known as pitch class profile (PCP) or chroma features, has been successfully used for music computation tasks such as chord and key estimation [40], [41], cover-song identification [29], [42], and music structure analysis [17].

For the computation of the chroma features we favor the approach proposed in [18], and briefly explained as follows.

First, the input signal is passed through a multi-rate, constant-Q bank of elliptical filters. The center frequency of each filter is defined as $f(p) = 440 \times 2^{(p-69)/(12)}$, where $p \in [1, 120]$ is the MIDI note number.[1] The filters are designed to have a quality (Q) factor of 25, and a transition band half the width of the passband. Forward–backward filtering is used to avoid group delays. Then, the local energy is computed on each of the resulting sub-band signals, using a rectangular moving window of length 4410 samples with 50% overlap. For a sampling frequency of $f_s = 22050$ Hz, this results in a rate of ten features. This process returns the short-term mean-square power value, $X(p, n)$ per pitch band p and analysis frame n, from which the frame-wise chroma feature vector can be calculated as

$$c_b(n) = \sum_{i=0}^{O-1} X(b + i\beta, n) \tag{1}$$

where $b \in [1, \beta]$ is the chroma bin number, and O is the total number of octaves considered. In this implementation, $\beta = 12$ and $O = 10$ octaves.

Chroma features are adept at characterizing harmonic information in the music signal, but are susceptible to noise introduced by short-term events. To alleviate this problem we test feature downsampling and the use of beat-synchronous feature blocks. Both these approaches help to smooth out the effect of short-term events such as transients and different types of pitch simultaneities (e.g., arpeggios, walking bass lines, trills), while at the same time reducing the dimensionality of the recurrence plot representation (N^2, where N is the total number of analysis frames). The beat-synchronous representation has the added benefit of minimizing the effect of tempo variations.

In this paper, we use the beat tracking approach introduced in [43]: given an onset detection function and an initial global tempo estimate, this method uses dynamic programming to find the sequence of beats that most closely follows the tempo, while at the same time having maximal or near-maximal value in the detection function. We use the implementation and parameter set used by the author for cover song identification.[2] After tracking, feature blocks are averaged between consecutive beats.

Additionally, we test the use of the *chroma energy normalized statistics* (CENS [18]). These features seek to introduce robustness against dynamic changes by quantizing each normalized chroma vector into five logarithmically distributed levels: more than 40% of total signal energy, between 20 and 40%, between 10 and 20%, between 5 and 10%, and less than 5%. To minimize the effect of short-term changes in the signal, the sequence of quantized features is convolved with a w-long Hann window and the output further downsampled by a factor d. In this implementation $w = 41$ and $d = 10$, resulting on a feature rate of 1 Hz.

A final variation, proposed in [44], is inspired by the MFCC widely used in speech and music processing applications. In MFCC analysis, the log, mel-frequency magnitude spectrum is transformed to the *cepstral* domain using the discrete cosine transform (DCT). The compression properties of the DCT mean that the spectral envelope is efficiently encoded by a few low coefficients in the cepstral domain, assumed to contain most timbral information in the audio signal. Conversely, we can filter out those coefficients, thus effectively de-emphasizing the timbre of the sound in favor of the "fast-changing" spectral components commonly related to pitch periodicities.

This intuition can be exploited in the computation of chroma features, by applying the DCT to $\log_{10}(\alpha \times X(p, n) + 1)$, where α is a constant, and then zeroing the λ-lowest DCT coefficients

[1] $p = 69$ corresponds to $A_4 = 440$ Hz, immediately above middle C.

[2] http://labrosa.ee.columbia.edu/projects/coversongs/

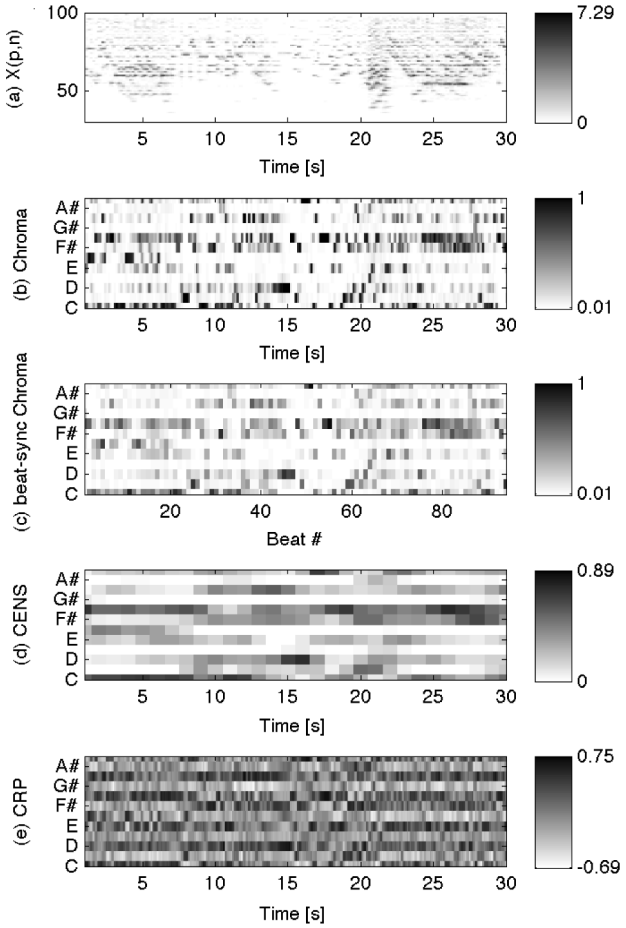

Fig. 2. Feature extraction for a 30-s excerpt of Mozart's Sonata for Piano # 1 in C major, performed by Glenn Gould. The figure shows the (a) short-term mean-square power values per pitch sub-band, (b) chroma, (c) beat-synchronous chroma, (d) CENS, and (e) CRP features.

before transforming back using the inverse DCT. The resulting representation is a *whitened* version of $X(p,n)$, from where a new timbre-less chroma vector can be derived using (1). These features are known as **c**hroma DCT-**r**educed log-**p**itch (CRP). In this study we use $\alpha = 1000$ and $\lambda = 55$.

Fig. 2 shows all feature types for an excerpt of a piano sonata by W.A. Mozart. In this paper, we use the Chroma Matlab toolbox[3] for feature extraction. All feature vectors are normalized to have unit length.

B. Recurrence Plots

Repetition, or recurrence, is an important feature of music, underpinning the organization of sounds into beats, bars, motives, and sections [45]. Recurrence is also a key property of complex dynamical systems and a wide variety of data series, making techniques used for their analysis also applicable to the analysis of music. This research focuses in the use of one such technique, the recurrence plot, as a representation of global structure in music signals.

A recurrence plot (RP) is a tool for the visualization and analysis of dynamical system as described by their delay-coordinate space, in which all possible states of the system are represented [24]. Commonly, RPs are binary matrices where a value of one in the ith row and jth column of the matrix indicates that the state of the system is the same (i.e., recurs) at positions i and j of the time series, $i, j \in [1, N]$.

We can reconstruct the delay-coordinate space for a given one-dimensional time series $x(t)$ using a process known as time-delay embedding. In this process we choose an embedding dimension m, and a time delay τ, such that each time t in the series is now represented by a vector obtained by concatenating the values $x(t), x(t-\tau), \ldots, x(t-(m-1)\tau)$. Likewise, we can apply the same process to a multi-dimensional time series such as the chroma sequence $C = \{c_b(n)\}$, where $n \in [1, N]$ and $b \in [1, 12]$. In this case, the nth time-embedded vector $\mathbf{x_c}(n)$ is simply the result of stacking the m, τ-spaced chroma vectors spanning the block from time $n-(m-1)\tau$ to n:

$$\mathbf{x_c}(n) = (c_1(n), c_1(n-\tau), \ldots, c_1(n-(m-1)\tau), \ldots \\ c_{12}(n), c_{12}(n-\tau), \ldots, c_{12}(n-(m-1)\tau))^{\mathrm{T}}. \quad (2)$$

The recurrence plot R for the sequence $X = \{\mathbf{x_c}(n)\}$ is computed such that $R(i,j) = 1$ if $\mathbf{x_c}(i)$ and $\mathbf{x_c}(j)$ are no farther than a distance ϵ from each other in the delay coordinate space, and $R(i,j) = 0$ otherwise. In this case, we are not identifying repeating frames, but repeating sequences of frames. This can be expressed as

$$R(i,j) = H(\epsilon - \| \mathbf{x_c}(i) - \mathbf{x_c}(j) \|), \\ \mathbf{x_c}(i) \in \mathbb{R}^{12\times m}, i, j = m, \ldots, N \quad (3)$$

where $\| \cdot \|$ is a norm (e.g., Euclidean norm), and H is the unit step function. The process is illustrated in Fig. 3.

To understand the effect of embedding, let us consider the normalized sequence C as a trajectory in the 12-dimensional chroma space. In this context, $R(i,j) = 1$ occurs when the jth point of the trajectory fall within the hypersphere of radius ϵ centered at the ith point. When m increases, however, $R(i,j) = 1$ can only occur if the sub-trajectory $c_b(j-(m-1)\tau : j)$ stays within the 12-dimensional ϵ-tube around the sub-trajectory $c_b(i-(m-1)\tau : i)$. In other words, the two sub-trajectories must be not only close but also parallel. This effectively reduces the amount of noise due to random crossings of the trajectory in the feature space, at the cost of missing recurrences of sections shorter than $(m-1)\tau$ samples.

The optimal choice of τ and its effect on the representation depends on context. For high feature rates, a choice of $\tau > 1$ can be used to effectively increase the length of the ϵ-tube without increasing the dimensionality of the embedding. Even for critically sampled features, if the information we aim to characterize happens to be separated exactly by a period of τ, then the resulting plot will be cleaner and more informative, e.g., a beat-synchronous feature set where the important information is in every other beat, or in the downbeat. However, if the sequence changes rapidly and semi-randomly, using $\tau > 1$ comes at the cost of noisier plots. It is worth noting that using $\tau = 1$ and $m > 1$ is equivalent to using the texture window—[4] or shingle-based [46] analyses found across the MIR literature.

From the above it is clear that, beyond embedding, the choice of ϵ has an important impact on the topology of the recurrence plot. In our basic implementation, RPs are computed using the Euclidean norm between vectors normalized to unit length after

[3]http://www.mpi-inf.mpg.de/~mmueller/chromatoolbox/

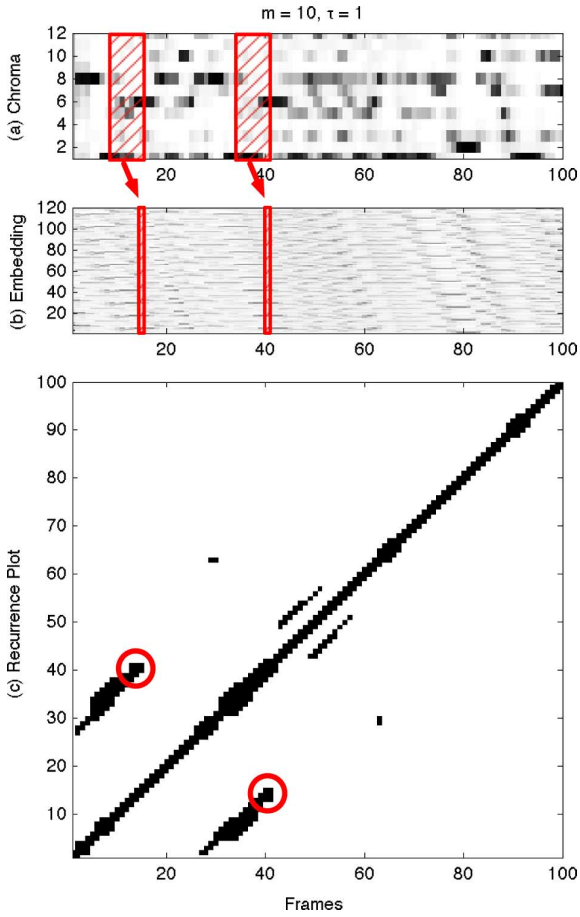

Fig. 3. Time-delay embedding with $m = 10$ and $\tau = 1$. (a) Blocks of ten consecutive chroma vectors are (b) stacked into single, 120-dimensional vectors. (c) A distance of less than ϵ between two embedded vectors at positions i and j, results in $R(i,j) = R(j,i) = 1$.

embedding. We will refer to this as the normalized Euclidean distance method (NEUC). In this case we define a threshold parameter $\theta \in [0,1]$, such that $\theta = \epsilon/2$. This seemingly arbitrary mapping helps maintain the range of θ constant across thresholding methods. A choice of lower θ results in a sparser RP, and viceversa.

Additionally, we will explore two alternative approaches. The first is known as the *fixed amount of nearest neighbors* (FAN) approach, where ϵ is varied for every n to ensure that $R(i,j) = 1$ for the $\theta \times N$ points closest to the ith point of the trajectory. In this case, the parameter θ controls the percentage of nearest neighbors. A variant of this method is favored in [29]. The second approach requires dynamic adjustments of ϵ to ensure that the recurrence rate $RR = (1/N^2)\sum_{i,j=1}^{N} R(i,j)$ is the same for all plots. In this case, $\theta =$ RR fixes the density of the plot as a percentage of its recurrences.

Our recurrence plot implementation is based on functions from the CRP Matlab toolbox.[4]

C. Similarity

As discussed in Section II-D, pairwise similarity between recurrence plots is measured using the normalized compression distance (NCD), to be briefly introduced in the following. For a comprehensive discussion the reader is referred to [47] and [48].

It can be shown that the information distance between two objects o_1 and o_2, up to a logarithmic additive term, is equivalent to

$$ID(o_1, o_2) = \max\{K(o_1 \,|\, o_2),\ K(o_2 \,|\, o_1)\} \tag{4}$$

where $K(\cdot)$ denotes the Kolmogorov complexity. The conditional complexity $K(o_1 \,|\, o_2)$ measures the resources needed by a universal machine to specify o_1 given o_2.

The information distance in (4) suffers from not considering the size of the input objects, and from the non-computability of $K(\cdot)$. These issues are addressed by normalizing the distance as

$$\mathrm{NID}(o_1, o_2) = \frac{\max\{K(o_1 \,|\, o_2),\ K(o_2 \,|\, o_1)\}}{\max\{K(o_1),\ K(o_2)\}} \tag{5}$$

and by approximating $K(\cdot)$ using $C(\cdot)$, the size in bytes of an object compressed using a standard compression algorithm. Using this principle, it can be shown that (5) is approximated by the normalized compression distance

$$\mathrm{NCD}(o_1, o_2) = \frac{C(o_1 o_2) - \min\{C(o_1),\ C(o_2)\}}{\max\{C(o_1), C(o_2)\}} \tag{6}$$

where $C(o_1 o_2)$ is obtained by compressing the concatenation of objects o_1 and o_2 [48]. In this paper, the objects are the recurrence plots of each track stored as binary files in row-major order, i.e., as a vector of consecutive rows. We use the NCD implementation in the *CompLearn* toolkit[5] with the bzip2 compression algorithm.

D. Why the NCD?

There are a number of reasons that motivate our choice of the NCD as distance function. First, its simplicity. Unlike the DP-based approaches discussed in Section II-B, the NCD does not require aligning the feature sequences being compared, a task that is significantly more complex for the comparison of 2-D representations such as recurrence plots. Instead it can be easily and robustly computed using efficient and widely available compression algorithms.

Second, the NCD has been shown to be quasi-universal in that it approximates all other similarity metrics provided the used method of compression is *normal* [47], [48]. Tellingly, in these and other references the NCD is oftentimes referred to as the *Universal Similarity Metric*. This quasi-universality means that the NCD is able to characterize the most salient similarities between objects, in a manner that is not specific to features or applications [47]. This versatility has been demonstrated by its successful application in fields as diverse as genomics, astronomy, language, the detection of plagiarism and, as mentioned in Section II, music. The reader is referred to [48] for a comprehensive list of examples.

Third, as discussed in Section II-A, we start from the assumption that most music consists of repetitions of a relatively small set of patterns. Thus, it seems appropriate to measure similarity

[4]http://www.agnld.uni-potsdam.de/~marwan/toolbox/

[5]http://www.complearn.org

as a function of the information overlap between patterns of repetition in musical recordings. In this paper, we have chosen to measure the distance between these patterns of repetition, as represented by the RPs, and not between the repetitive patterns themselves, which are more difficult to extract than the former (see [49] for a review).

Finally, there is prior art suggesting that the pairing of recurrence plots and the NCD can be successfully used for information retrieval, at least in the context of protein structure comparison [33], [50]. In this paper, we document our attempts to test if this is also true of music, and under which conditions.

IV. Experimental Setup

A. Task

The goal of the proposed approach is to retrieve music recordings according to their structural similarity to an audio query. One way to evaluate success is by means of a known-item search on a music database containing pieces with common structure. However, an unambiguous categorization of structural types in music is far from trivial [51], making the collection of thousands of these recordings a difficult and time consuming task requiring significant input from musicians and musicologists.

Therefore, rather than embarking in a complex annotation process, we evaluate our approach using an approximation to pure structure-based retrieval: the retrieval of multiple interpretations of works from the classical repertoire. Classical music is used as, in this tradition, the structure of different performances can be expected to be consistent. Admittedly, different interpretations of a music piece are expected to have more than structure in common, potentially introducing a bias in the evaluation process. In an attempt to minimize this effect, the data sets are chosen to include significant variations in tempo, dynamics, ornamentation, recording conditions, and, in a few cases, the local structure, across different interpretations. Globally, they are composed of recordings using similar instrumentation (piano, orchestra), to increase the likelihood of inter-work confusions and emphasize the difference with texture-based similarity approaches. We believe that both the positive and negative results presented in the following show that, despite its caveats, the chosen task is indeed evaluating the system's ability to measure structural similarity.

B. Data Sets

We perform retrieval on two collections of multiple performances of works from the classical and romantic period. All files are 128 kb/s MP3s with sampling frequency of 44.1 kHz.

The first collection, to be referred to as the *training set*, consists of 123 recordings of 19 works by eight composers: Beethoven (5), Berlioz (1), Brahms (2), Chopin (2), Mahler (2), Mendelssohn (1), Mozart (5), and Tchaikovsky (1). Eight of those works are excerpts of piano compositions, with 3 to 13 renditions each and lengths ranging from 1 to 8 minutes. The resulting 56 recordings were done between 1946 and 1998, and include performances by 25 famous pianists.[6] The remaining 11 works are symphonic movements including six renditions per movement (seven in one case) for a total of 67 tracks recorded between 1948 and 2008, and featuring 34 different conductors. Durations range from 3 to 10 minutes long. This set is used to find the best combination of system's parameters for the proposed approach.

The second collection, the *testing set*, contains 2919 recordings of 49 Chopin's Mazurkas for solo piano, inspired by the Polish folk dance.[7] The average number of renditions per Mazurka is 58, with group sizes ranging between 41 and 95. The earliest recording in the collection is from 1902, and the newest from 2008, featuring 135 different pianists. Accurate retrieval in this collection is further complicated by the similarities in style and instrumentation among all recordings. The test set is used to assess the approach's robustness to noise and scale, and to validate the results obtained using the training set.

C. Methodology

The evaluation consists of a known-item search, whereby all recordings are used as queries and success is measured by the ability of the system to return all renditions of a given work at the top of a ranked list. This is quantified using the mean average precision.

Let us assume a set $\mathcal{Q} = \{q_i\}, i \in [1, K]$, of performances of a given work. $\mathcal{Q}$ is a subset of the M-long collection $\mathcal{D}$, such that $M \gg K$. For a given query q_i, we can create a ranked list $\mathcal{L}(r)$ by organizing $\mathcal{D}$ in ascending order according to the NCD between all items and q_i, such that $r \in [1, M]$ is the rank. The average precision can thus be defined as

$$\mathrm{AP} = \frac{1}{K} \sum_{r=1}^{M} P(r)\Omega(r) \tag{7}$$

where $P(r)$ is the precision at a given rank r

$$P(r) = \frac{1}{r} \sum_{j=1}^{r} \Omega(j) \tag{8}$$

and $\Omega(r)$ is a binary function characterizing the relevance of the item at rank r

$$\Omega(r) = \begin{cases} 1, & \text{if } \mathcal{L}(r) \in \mathcal{Q} \\ 0, & \text{otherwise.} \end{cases} \tag{9}$$

The mean average precision (MAP), which will be reported in all experiments, is simply the mean of AP values across all queries in a given set. Significance testing was performed using Friedman's ANOVA, and post-hoc multiple comparison analysis using Tukey's range test with confidence coefficient of 95% [53]. This is consistent with common practice in the MIR evaluation exchange (MIREX).[8]

[6]These recordings were collected at the Austrian Institute for Artificial Intelligence for the work in [52].

[7]These recordings were collected as part of the *Mazurka* project at the Center for the History and Analysis of Recorded Music, London, U.K.

[8]http://www.music-ir.org/mirex/wiki/

V. Results and Discussion

The proposed implementation can be summarized as follows.

- Compute the feature sequence F_f^N for a given music signal. In our experiments, the feature type F can take the values C (chroma), CENS or CRP; the feature rate f can be beat-synchronous (beat), the default 10 features/s, or downsampled to 5, 2.5, 1.25, 1, 0.5, or 0.333 features/s; and the length of the feature sequence N can vary according to signal length (var) or be resampled to a fixed value of 300, 500, 700, 900, or 1100 frames. For example, using beat-synchronous CRP features with variable length is denoted as $\mathrm{CRP}_{\mathrm{beat}}^{\mathrm{var}}$.
- Reconstruct the delay coordinate space using time-delay embedding dimension m and delay time τ. The experiments use monotonically increasing values of $m \in [1,7]$ and $\tau \in [1,5]$.
- Generate the recurrence plot using the thresholding method T_θ, where T can be the normalized Euclidean (NEUC), fixed amount of nearest neighbors (FAN) or fixed recurrence rate (RR) method; and the thresholding value $0.05 \leq \theta \leq 0.95$.
- Once this process is repeated for all signals, we compute the pairwise normalized compression distance between RPs, using bzip2 compression.

The following discusses the effect of these variables in the retrieval of multiple performances of a given work according to structural similarity.

A. Feature Processing

The first column of Table I compares the effect of using beat-synchronous segmentation versus feature downsampling for $\mathrm{C}_f^{\mathrm{var}}, m = \tau = 1$ and $\mathrm{NEUC}_{0.5}$. Experiments using f =10 or 5 features/s are skipped to avoid computation and storage problems for large audio files. Tests are performed on the training set.

At first glance, these results show beat-synchronous analysis outperforming downsampling, with a maximum $\mathrm{MAP} = 0.364$. However, the low performance represented by these values is indicative of problems of more significance than this comparative advantage.

From the discussion in Section II-D, we can recall that, for the NCD to work correctly, the used compressor must be normal. This, among other conditions, means that it should be invariant to the length of the objects being compared. Ref. [54] demonstrates that this is not necessarily true for most well-known compressors, including bzip2, resulting in a distance that is skewed by the objects' size. This problem can be alleviated by setting the compressor parameters to favor quality over speed of the compression, or by using a predictive compression algorithm, such as PPM [55], that is free from block-based analysis. However, our experiments are already performed using the highest-quality settings of bzip2, and unreported tests using PPM did not improve performance on the training set.

A simple solution to this problem is to resample all feature sequences to a fixed length N, using decimation or interpolation after applying an anti-aliasing filter, thus eliminating the variability in RP size. Results for a range of fixed N values are reported in the remaining columns of Table I. The resampling strategy improves performance over the variable-length case, with best $\mathrm{MAP} = 0.630$ for C_5^{700}, although all results in the gray-shaded area of the table are statistically the same. Fig. 4 shows the improvement caused by resampling to be significant, by means of a multiple comparison analysis using Tukey's range test [53], where it can be observed that the comparison intervals of the first two estimates, corresponding to $\mathrm{C}_{\mathrm{beat}}^{\mathrm{var}}$ and C_5^{700}, respectively, do not overlap. This difference in performance underpins the effect that object size has on the NCD.

TABLE I
Chroma Features: Results on the Training Set for Variations of Feature Rate f and Sequence Length N

		N				
f	*var*	300	500	700	900	1100
beat	0.364	0.499	0.482	0.496	0.462	0.441
10	-	0.449	0.424	0.444	0.394	0.359
5	-	0.541	0.598	**0.630**	0.629	0.588
2.5	0.188	0.506	0.518	0.576	0.612	0.598
1.25	0.254	0.386	0.514	0.549	0.576	0.547
1	0.201	0.225	0.290	0.511	0.544	0.546
0.5	0.183	0.244	0.268	0.473	0.467	0.476
0.333	0.182	0.252	0.264	0.446	0.457	0.451

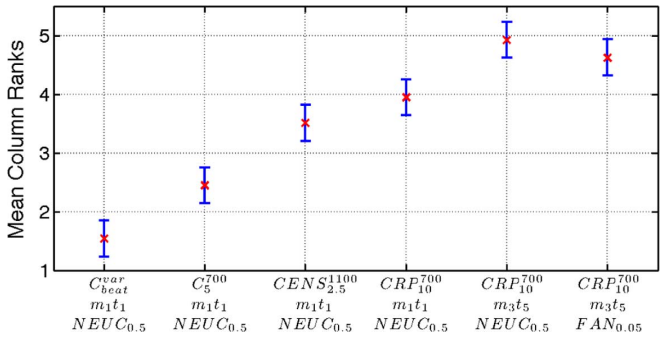

Fig. 4. Significance testing: post-hoc multiple comparison analysis using Tukey's range test with confidence coefficient of 95%. Results for six parameter combinations show test estimates (x) and comparison intervals around them.

B. Beat Tracking

Notably, while improving on the variable-N scenario, performance using beat-synchronous segmentation quickly reaches an upper-bound slightly lower than $\mathrm{MAP} = 0.5$. This is mostly due to inconsistencies in beat tracking.

It has been extensively discussed in the literature that the most common issue in automatic beat tracking are the so-called octave errors, whereby the tempo of a piece of music is estimated at half or twice its actual rate (see [56] for a review). This doubling/halving of the beat rate is emphasized by expressive changes of tempo, variations of meter, non-percussive instrumentation, and the lack of a strong beat, all of which are commonly found in orchestral and piano classical performances such as the ones in the training set.

To exemplify this, Fig. 5 compares the recurrence plots of two performances of the 3rd movement of Mozart's Symphony No. 40 in G minor, KV. 550. It can be seen that the number of tracked beats in the first performance (identified as Brueggen1985) is twice as high as the number of beats identified in the second performance (Brueggen1991), quadrupling the size of the corresponding RP. This produces a smoothing out of the finer, short-term structure in the second plot, which looks like a low-pass

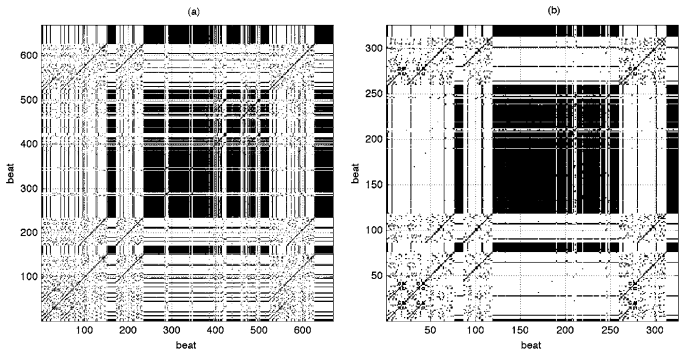

Fig. 5. Comparison of recurrence plots for two performances of W. A. Mozart's Symphony # 40, movement 3. The figures illustrate how beat-tracking inconsistencies lead to dissimilarities in the RP.

TABLE II
CENS FEATURES: RESULTS ON THE TRAINING SET FOR VARIATIONS OF FEATURE RATE f AND SEQUENCE LENGTH N

	N				
f	300	500	700	900	1100
10	0.776	0.788	0.816	0.820	0.813
5	0.762	0.789	0.817	0.817	0.813
2.5	0.767	0.785	0.813	0.821	**0.827**
1.25	0.748	0.811	0.816	0.823	0.824
1	0.395	0.411	0.811	0.808	0.814
0.5	0.398	0.398	0.795	0.801	0.791
0.333	0.357	0.379	0.733	0.726	0.750

TABLE III
CRP FEATURES: RESULTS ON THE TRAINING SET FOR VARIATIONS OF FEATURE RATE f AND SEQUENCE LENGTH N

	N				
f	300	500	700	900	1100
10	0.783	0.847	**0.863**	0.846	0.844
5	0.789	0.848	0.852	0.848	0.849
2.5	0.738	0.829	0.849	0.832	0.821
1.25	0.683	0.741	0.797	0.785	0.762
1	0.319	0.397	0.744	0.733	0.727
0.5	0.331	0.331	0.589	0.615	0.622
0.333	0.303	0.316	0.559	0.544	0.554

filtered version of the first. Fixing N after beat tracking does nothing to address the resulting difference in topology, which is bound to increase the distance between the plots and therefore reduce the accuracy of retrieval. Octave errors are unavoidable, even with state-of-the-art beat-tracking systems such as the one used. Thus, the remainder of these experiments are performed without beat-synchronous analysis.

C. Feature Type

Tables II and III, compare performance over variations of N and f for the two chroma variants discussed in Section III-A: CENS and CRP, respectively. Both sets of results show improvement over the use of chroma features. In the case of the CENS features, each N, f combination sees an average net improvement of 0.26 in MAP, with a 0.197 increase in best performance for CENS_4^{1100}. For CRP features, the net increase in best performance, for CRP_{10}^{700}, is of 0.233 upon chroma features and 0.036 upon CENS features. Fig. 4 shows the differences with chroma features to be significant, while the differences between CRP and CENS features are not.

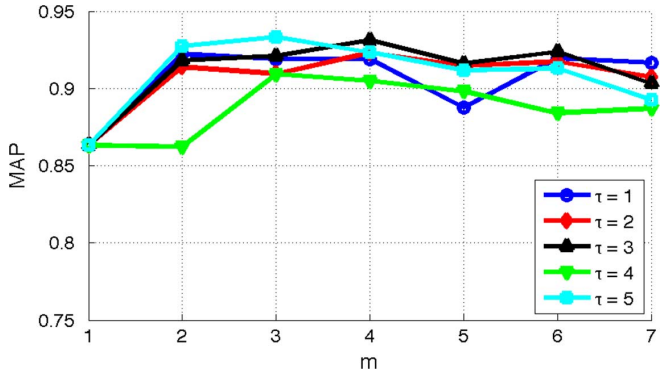

Fig. 6. Retrieval accuracy on the training set for variations of embedding dimension m and delay τ.

In both tables, best results (gray shaded) tend to appear towards the top-right corner and worst results are mostly concentrated on bottom-left cells. The trend shows that, while smoothing is necessary to reduce the effect of short-term events, excessive amounts produced by using low N and f values have a negative impact on retrieval. Notably, results obtained using CENS features, where downsampling is an active part of their design, are best for maximum N and mid-range f values. Conversely, CRP results are best for maximum f (no downsampling) and mid-range N values.

More importantly, the removal of timbral content partly achieved through quantization in the CENS, or via the zeroing of low DCT coefficients in the CRP, has the largest positive impact on performance. We conjecture that, at least for the training set, structural similarity is better characterized by harmonic than by timbral information in the signal, and that the "coloring" introduced by the sound's spectral envelope is unwanted for this task. This is consistent with preliminary experiments where chroma features outperform MFCCs for structural-based similarity [7], and with the wide use of chroma features for music structure analysis in the literature.

D. Recurrence Plot

Besides feature extraction, the choice of time-delay embedding variables, and the strategy used in the computation of the recurrence plots, have an impact on the accuracy of retrieval. Fig. 6 depicts results in the testing set using $1 \leq m \leq 7$ and $1 \leq \tau \leq 5$, for CRP_{10}^{700} and $\mathrm{NEUC}_{0.5}$.

The leftmost point of all curves correspond to a MAP of 0.863 for $m = 1$. It is readily observed that choosing $m > 1$ brings about and increase of MAP, with the majority of results in the [0.90, 0.94] range. Most curves peak somewhere between $2 \leq m \leq 4$, showing a slight decline after that. Best performance of $\mathrm{MAP} = 0.933$, for $\tau = 5$ and $m = 3$, is significantly better than for the case of no embedding (see Fig. 4).

The trend, however, is far from clear for variations of τ, with odd delay-values showing better performance for most m. Unlike in [29], we fail to see a clear correspondence between results obtained using same values of $(m-1)\tau$, e.g., for $[m = 2, \tau = 4]$ and $[m = 5, \tau = 1]$. This may be due to the small size of our training set, which creates the distinct possibility of overfitting. However, it is worth noting that variations of τ for a given m result in statistically insignificant differences of less

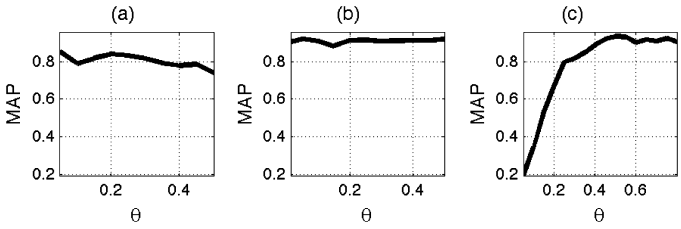

Fig. 7. Retrieval accuracy on the training set for different recurrence plot (RP) computation strategies and threshold values θ. Results are reported for the (a) fixed recurrence rate, (b) fixed amount of nearest neighbors, and (c) normalized Euclidean methods.

than 0.03. The only exception is the $\tau = 4$ curve which is notably lower. Since features are not event-synchronous and the sequence is resampled to a fixed length, there is no musical or temporal meaning to this delay value that can be used to explain this singularity. It can only be speculated that for a number of recordings in this particular dataset, $\tau = 4$ leads to a loss of smoothness in the embedded sequence that negatively affects the characterization of recurrence. It must be noted that at its worst, this delay value leads to results at the level of no embedding.

Fig. 7 compares RP computation strategies on the training set for optimal values CRP_{10}^{700}, $m = 3$ and $\tau = 5$. The subplots correspond to variations of the threshold θ using the (a) RR, (b) FAN, and (c) NEUC methods.

We can identify several trends in these plots. Overall, performance using RR is significantly lower than that obtained for FAN or NEUC for most values of θ. Both RR and FAN show best performance for lower threshold values, with maximum MAP for $\theta = 0.05$ equal to 0.854 and 0.921, respectively. RR performance decreases steadily with increases of θ, while FAN results on a much flatter response, indicating robustness against the choice of threshold value.

Conversely, NEUC performs best for larger θ values. As the steepness of the curve illustrates, NEUC is highly sensitive to the threshold choice, with $\theta \leq 0.2$ resulting on $\mathrm{MAP} < 0.8$ and $0.45 \leq \theta \leq 0.55$ resulting on $\mathrm{MAP} \geq 0.92$.

As can be seen in Fig. 4, there is no statistically significant difference in best MAP between using NEUC and FAN. FAN was used in the original definition of RP [23], and has been reported to be particularly well suited for cross-recurrence analysis of noisy data [29], [57]. However, since the number of recurrences per column is fixed, the plot is non-symmetrical, meaning that we have to store all N^2 values of the representation. On the other hand, the symmetry of NEUC means that only storing $(N^2 - N)/2$ values per plot is necessary, resulting in important memory and computational savings.

It can be argued that optimizing features and embedding for the default $\mathrm{NEUC}_{0.5}$ introduces an unfair advantage in these tests. However, a limited test of FAN for all possible combinations of m, τ and θ did not bring any improvements. Furthermore, results in Section V-F show how retrieval on the testing set is largely unaffected by the choice of RP thresholding strategy.

Finally, it is worth noting that testing all possible variable combinations is unfeasible, as it will result on more than 100 000 experiments on the training set (each including 123 runs). To put this in context, there are less than 600 such experiments mentioned in this paper (whether fully reported or not), accounting for more than 3 weeks of computation time.

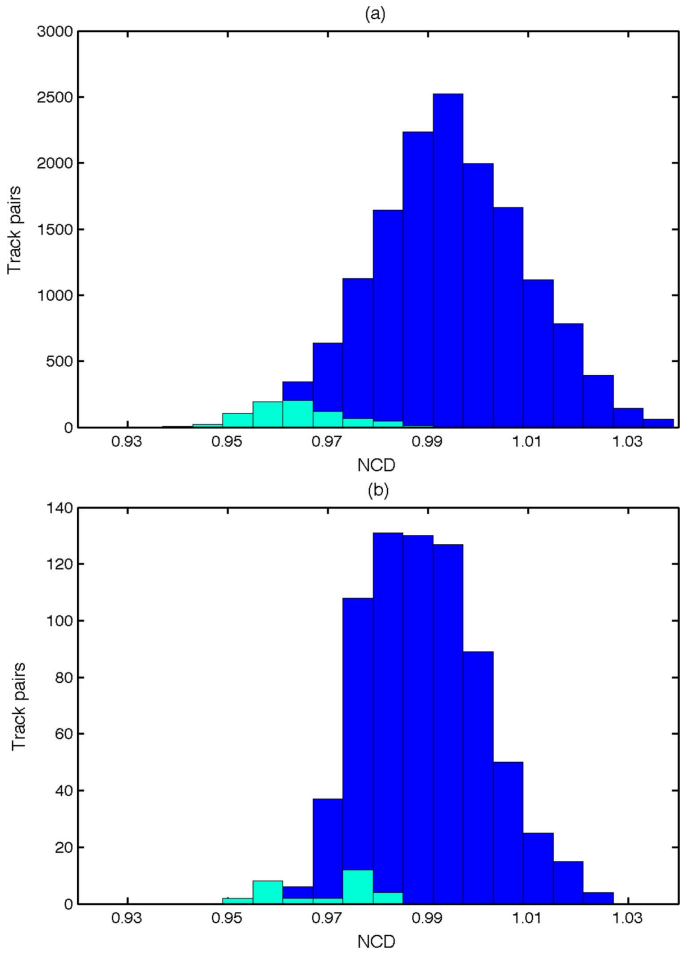

Fig. 8. Histogram of pairwise normalized compression distance (NCD) values for the training set using (a) all 123 tracks as queries, and (b) the six performances of the 4th movement of Berlioz's "Symphonie Fantastique" as queries. Light/dark blue (light/dark gray) coloring corresponds to counts of relevant/irrelevant track pairs.

E. NCD Behavior

Fig. 8(a) shows the histogram of pairwise NCD values for the training set using all 123 tracks as queries and not including self distances. The histogram is computed using the optimal parameter set CRP_{10}^{700}, $m = 3$, $\tau = 5$, and $\mathrm{NEUC}_{0.5}$. Counts of relevant track pairs are colored light blue, while those of irrelevant pairs are in dark blue.

The figure clearly illustrates the low resolution of the NCD, i.e., the extreme concentration of object distances in a small region of the function range, one of a number of artifacts caused by the use of standard compressors to approximate the Kolmogorov complexity [48]. In fact, our experiments showed all non-diagonal distances fall within the [0.94,1.04] range (where $\mathrm{NCD} > 1$ is another such artifact). Regardless of range, the ideal distribution should be bimodal, with each mode dominated by relevant/irrelevant pairs respectively, thus guaranteeing good discrimination between these groups. Instead, the observed distribution has a single mode, strongly dominated by irrelevant

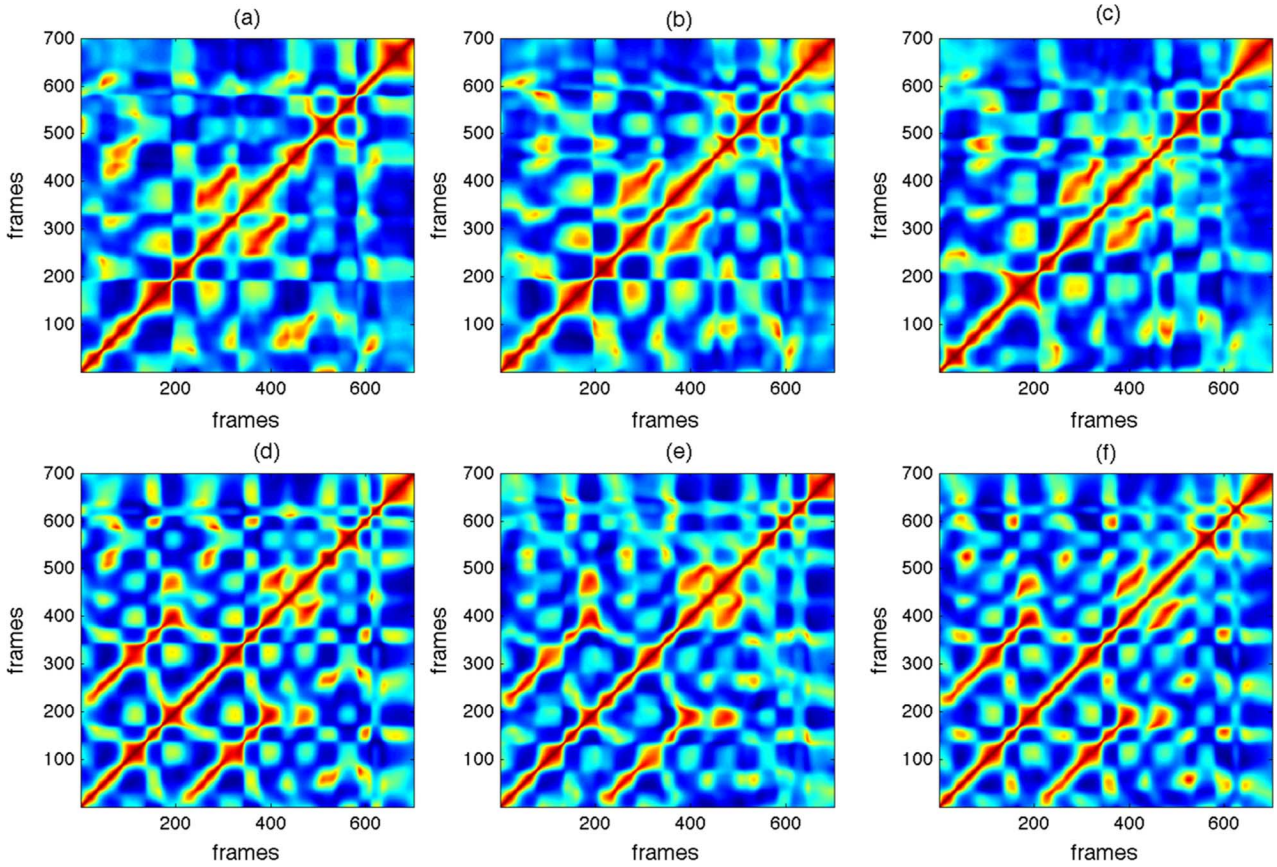

Fig. 9. Comparison of distance plots for six performances of Hector Berlioz's "Symphonie Fantastique", movement 4. The upper plots (a)–(c) represent an AB structure, while the lower plots (d)–(f) represent an AAB structure (the repetition is clearly represented by the long side diagonals starting around frame 200). The difference, resulting from conductors choosing to omit (or include) a long repeated section at the beginning of the movement, causes the proposed approach to characterize these recordings as dissimilar.

pairs, which are majority, and a strong concentration of relevant pairs in its lower tail. The portion of relevant pairs that tails into the overall distribution accounts for most retrieval errors, keeping MAP below 0.94.

When limiting queries to performances of a given work, most resulting histograms mirror Fig. 8(a), with informal tests unable to identify trends related to changes of tempo or recording conditions (both of which vary widely across performances). The one notable exception is illustrated by the example in Fig. 9, which shows distance plots for the six performances of the 4th movement of Berlioz's "Symphonie Fantastique". The score of this piece includes a repetition of the first 77 bars of this movement before entering its second half, roughly describing an AAB structure (visible in the bottom three plots of the figure). The top three performances in the figure (plots a–c), however, ignore that repetition resulting on a shorter AB structure. In this subset, this structural difference accounts for an important decrease in average precision. This is illustrated by the histogram in Fig. 8(b), showing pairwise NCD values when the six performances in this subset are used as queries. Notably, the distribution of relevant pairs [light blue (light gray)] is bimodal, with modes alternating between the upper and lower performance groups in Fig. 9, depending on the query.

This result shows that the current approach in general, and the NCD in particular, while able to characterize *global* structural similarity, is not able to cope with strong structural variations, such as repeating or omitting entire sections. One potential reason is that the application of the NCD, which was mostly designed for the comparison of string data, to 2-D representations is not straightforward and depends on the method used for the scanning of the RP. It is possible that there are better solutions than the simple row-major order scanning used in this implementation, although informal experiments with other methods resulted in no improvement.

A more likely cause, however, is that this shortcoming stems from our choice to characterize similarity between patterns of repetition, as represented by the recurrence plot. In this context, omitting a section effectively changes the RP in ways that justify the resulting dissimilarity. A potential solution that could be robust to such changes is to account for the similarity between the repetitive patterns themselves. However, the sequential [18] or joint [13] estimation of these patterns and their activations is far from trivial, remaining beyond the scope of this paper.

F. Noise Robustness and Validation

In Section II-D, we discuss a number of applications of the NCD including measuring the similarity of protein structures using contact maps [33] and grouping MIDI files according to composer and genre [48]. Both these works use small datasets that include only a few tens of objects, a common shortcoming in much of the NCD literature.

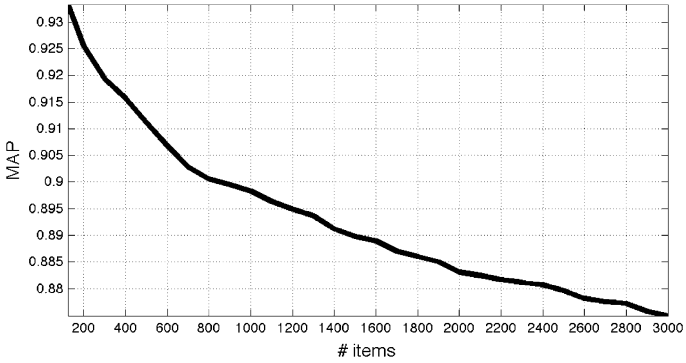

Fig. 10. Retrieval accuracy versus data set size. Tests are performed on the training set plus increasingly larger sections of the testing set.

However, there are indications in the literature that the ability of the NCD to robustly characterize similarity is compromised for large datasets. For example, the authors [48] report that increasing the size of the collection (to more than 40 items) results in increasingly distorted clustering trees, a problem they ascribe to the low resolution of the NCD matrix discussed above. In [58], the authors go one step further, questioning the robustness of the NCD when faced with large data sets, and showing its inadequacy at grouping similar protein structures on a set of 2771 objects.

In order to address this concern, we perform two tests to evaluate the proposed approach on a large dataset. For the first test we assess the accuracy of training set retrieval on a collection including progressively larger sections of the test set as noise. Fig. 10 depicts MAP versus collection size values starting at 123 (training set size) and finishing at 3000 (training plus testing sets). The results are computed using the optimal parameter set CRP_{10}^{700}, $m = 3$, $\tau = 5$, and $\mathrm{NEUC}_{0.5}$. The plot clearly shows that performance is negatively affected by the increasing amount of noise. The decrease in MAP is steep at first, but less pronounced for collection sizes of 1600 items or more. MAP for the full combined set is 0.875. The results at both ends of the curve, while statistically different, are only 0.058 apart demonstrating robust retrieval on a larger, noisier set.

Second, we validate the performance of the optimal system using the 2919 tracks in the test set as queries, resulting on $\mathrm{MAP} = 0.767$. Using $\mathrm{FAN}_{0.05}$, instead of $\mathrm{NEUC}_{0.5}$, results on a similar $\mathrm{MAP} = 0.764$, further confirming the equivalency of these two approaches. The high MAP values obtained on the test set validate the optimality of the chosen parameters, and demonstrate that successful retrieval is not specific to the training set. Furthermore, the approach is shown to be robust for a complex data set of thousands of piano performances, one in which grouping and discrimination cannot be explained by sonic or textural characteristics in the recordings, as is the case for most previous similarity research in MIR.

Finally, concerns about the scalability of NCD-based retrieval are successfully addressed by the last two tests. However, the difference with previous large-scale tests in the literature suggest that the metric is not as invariant to object representation as previous works suggest. In our case, it can be argued that good results are as much the consequence of using NCD as a metric, as they are of the choice of recurrence plots of (length-normalized) CRP features as signal representations.

VI. Conclusion

This paper presents a new method for the characterization of the structural similarity between music audio recordings. In contrast to previous work on music structure analysis in MIR, the proposed approach does not concentrate on segmenting and clustering sections, but in measuring the distance between intermediate representations of the signal's structure. It represents music signals using recurrence plots (RP) of chroma feature sequences, and characterizes pairwise similarity using the normalized compression distance (NCD). This combination permits the characterization of similarity despite changes in tempo, instrumentation and global key, as long as the overall structure remains unchanged. The choice of NCD provides an alternative metric of music similarity that is simple to compute, versatile and robust. Furthermore, experimental results contribute an optimal parameterization of the proposed approach.

More specifically, our evaluation shows that 1) chroma DCT-reduced log-pitch (CRP) and chroma energy normalized statistics (CENS) features improve performance upon standard chroma features, by minimizing the effect of fast signal events and the coloring introduced by different ensembles and instrumental sounds, 2) the potential advantages introduced by beat-synchronous analysis are offset by beat-tracking inconsistencies that artificially increase the distance between similar songs, 3) normalizing the length of the feature sequences to a fixed, pre-defined value is necessary to redress the negative impact that variable signal lengths have on the NCD computation, and 4) the use of time-delay embedding reduces the detection of spurious recurrences, generally resulting in better retrieval performance. Notably, the results in Section V-F show that good performance is not restricted to the training dataset, and that the NCD can be successfully used for retrieval on collections of a few thousand recordings.

It is worth noting, however, that our results only indicate robustness for the characterization of global structural similarity. Experimental results show that the approach is sensitive to strong structural changes. This limits the potential use of NCD-based similarity in the modeling of the relationships that exist between recordings featuring such structural changes, e.g., covers or remixes. Future work will explore solutions to this problem based on computing independent RPs from *shingles* of the feature sequence [46], and using alternative ways of scanning trough the RP representation. We will also investigate ways to measure the similarity between repetitive patterns extracted using the method in [13].

Acknowledgment

The author would like to thank G. Widmer, W. Goebl, and C. Sapp for kindly providing access to their music datasets; R. Marwan, D. Ellis, R. Cilibrasi, and M. Müller for generously sharing their code; A. Glennon for proof-reading; and J. Serrà for his advice regarding the RP literature and helpful comments about this paper.

References

[1] E. Pampalk, "Computational models of music similarity and their application to music information retrieval," Ph.D. dissertation, Vienna Univ. of Technol., Vienna, Austria, 2006.

[2] S. Essid, "Automatic classification of audio signals: Machine recognition of musical instruments," Ph.D. dissertation, Univ. Pierre et Marie Curie, Paris, France, 2006.

[3] J. Aucouturier, "Ten experiments on the modelling of polyphonic timbre," Ph.D. dissertation, Univ. of Paris 6, Paris, France, 2006.

[4] G. Tzanetakis and P. Cook, "Musical genre classification of audio signals," *IEEE Trans. Speech Audio Process.*, vol. 10, no. 5, pp. 293–302, Jul. 2002.

[5] F. Lerdahl and R. Jackendoff, *A Generative Theory of Tonal Music*. Cambridge, MA: MIT Press, 1983.

[6] D. Huron, *Sweet Anticipation: Music and the Psychology of Expectation*. Cambridge, MA: MIT Press, 2006.

[7] J. P. Bello, "Grouping recorded music by structural similarity," in *Proc. Int. Conf. Music Inf. Retrieval (ISMIR-09)*, Kobe, Japan, 2009.

[8] R. B. Dannenberg and N. Hu, "Discovering musical structure in audio recordings," in *Proc. Int. Conf. Music Artif. Intell.*, Edinburgh, U.K., 2002, pp. 43–57.

[9] B. Logan and S. Chu, "Music summarization using key phrases," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Istanbul, Turkey, 2000, pp. 749–752.

[10] J.-J. Aucouturier and M. Sandler, "Finding repeating patterns in acoustic musical signals," in *Proc. 22nd Int. AES Conf. Virtual, Synthetic Entertainment Audio*, Espoo, Finland, 2002, pp. 412–421.

[11] G. Peeters, A. L. Burthe, and X. Rodet, "Toward automatic music audio summary generation from signal analysis," in *Proc. Int. Conf. Music Inf. Retrieval*, Paris, France, 2002, pp. 94–100.

[12] S. A. Abdallah, K. Noland, M. B. Sandler, M. Casey, and C. Rhodes, "Theory and evaluation of a Bayesian music structure extractor," in *Proc. Int. Conf. Music Inf. Retrieval*, London, U.K., 2005, pp. 420–425.

[13] R. J. Weiss and J. P. Bello, "Identifying repeated patterns in music using sparse convolutive non-negative matrix factorization," in *Proc. Int. Conf. Music Inf. Retrieval (ISMIR-10)*, Utrecht, The Netherlands, 2010, pp. 123–128.

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

[15] J. Foote and S. Uchihashi, "The beat spectrum: A new approach to rhythm analysis," in *Proc. IEEE Int. Conf. Multimedia Expo*, Tokyo, Japan, 2001, pp. 881–884.

[16] M. A. Bartsch and G. H. Wakefield, "Audio thumbnailing of popular music using chroma-based representations," *IEEE Trans. Multimedia*, vol. 7, no. 1, pp. 96–104, Feb. 2005.

[17] M. Goto, "A chorus-section detecting method for musical audio signals," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Hong Kong, 2003, pp. V-437–V-440.

[18] M. Müller, *Information Retrieval For Music and Motion*. Secaucus, NJ: Springer-Verlag, 2007.

[19] J. Paulus and A. Klapuri, "Music structure analysis using a probabilistic fitness measure and a greedy search algorithm," *IEEE Trans. Audio, Speech, Lang., Process.*, vol. 17, no. 6, pp. 1159–1170, Aug. 2009.

[20] J. Foote, "Arthur: Retrieving orchestral music by long-term structure," in *Proc. Int. Conf. Music Inf. Retrieval*, Plymouth, MA, 2000.

[21] T. Izumitani and K. Kashino, "A robust musical audio search method based on diagonal dynamic programming matching of self-similarity matrices," in *Proc. 9th Int. Conf. Music Inf. Retrieval (ISMIR'08)*, Philadelphia, PA, 2008, pp. 609–613.

[22] B. Martin, M. Robine, and P. Hanna, "Musical structure retrieval by aligning self-similarity matrices," in *Proc. Int. Conf. Music Inf. Retrieval (ISMIR'09)*, Kobe, Japan, 2009, pp. 483–488.

[23] J.-P. Eckmann, S. O. Kamphorst, and D. Ruelle, "Recurrence plots of dynamical systems," *Europhys. Lett.*, vol. 4, no. 9, pp. 973–977, Nov. 1, 1987.

[24] N. Marwan, M. C. Romano, M. Thiel, and J. Kurths, "Recurrence plots for the analysis of complex systems," *Phys. Rep.*, vol. 438, no. 5-6, pp. 237–329, Jan. 2007.

[25] J. D. Reiss and M. Sandler, "Nonlinear time series analysis of musical signals," in *Proc. Int. Conf. Digital Audio Effects (DAFx-03)*, London, U.K., 2003, pp. 24–28.

[26] E. Métois, "Musical gestures and embedding synthesis," Ph.D. dissertation, Mass. Inst. of Technol., Cambridge, MA, 1996.

[27] B. Schoner, C. Cooper, C. Douglas, and N. Gershenfeld, "Data-driven modeling of acoustical instruments," *J. New Music Res.*, vol. 28, pp. 28–2, 1999.

[28] M. Grachten, W. Goebl, S. Flossmann, and G. Widmer, "Phase-plane representation and visualization of gestural structure in expressive timing," *J. For New Music Res.*, vol. 38, no. 2, pp. 183–195, 2009.

[29] J. Serrá, X. Serra, and R. G. Andrzejak, "Cross recurrence quantification for cover song identification," *New J. Phys.*, vol. 11, Sep. 2009, article 093017.

[30] E. L. Sonnhammer and J. C. Wootton, "Dynamic contact maps of protein structures," *J. Mol. Graph. Modelling*, vol. 16, no. 33, pp. 1–5, 1998.

[31] A. Caprara, R. Carr, S. Istrail, G. Lancia, and B. Walenz, "1001 optimal PDB structure alignments: Integer programming methods for finding the maximum contact map overlap," *J. Comput. Biol.*, vol. 11, no. 1, pp. 27–52, Jan. 2004.

[32] W. Xie and N. V. Sahinidis, "A branch-and-reduce algorithm for the contact map overlap problem," *Res. Computational Biol. (RECOMB'06), Lecture Notes Bioinformatics*, vol. 3909, pp. 516–529, 2006.

[33] N. Krasnogor and D. A. Pelta, "Measuring the similarity of protein structures by means of the universal similarity metric," *Bioinformatics (Oxford)*, vol. 20, no. 7, pp. 1015–1021, 2004.

[34] R. Cilibrasi, P. Vitanyi, and R. de Wolf, "Algorithmic clustering of music based on string compression," *Comput Music J* vol. 28, no. 4, pp. 49–67, Dec. 2004 [Online]. Available: http://homepages.cwi.nl/paulv/papers/music.pdf

[35] M. Li and R. Sleep, "Genre classification via an LZ78 string kernel," in *Proc. Int. Conf. Music Inf. Retrieval*, London, U.K., 2005, pp. 252–259.

[36] M. Helén and T. Virtanen, "A similarity measure for audio query by example based on perceptual coding and compression," in *Proc. Int. Conf. Digital Audio Effects (DAFx'07)*, Bordeaux, France, 2007, pp. 173–176.

[37] T. Ahonen and K. Lemström, "Identifying cover songs using normalized compression distance," in *Proc. Int. Workshop Mach. Learn. Music (MML'08)*, Helsinki, Finland, 2008 [Online]. Available: http://www.dtic.upf.edu/rramirez/MML08/abstracts.pdf

[38] R. Shepard, "Circularity in judgements of relative pitch," *J. Acoust. Soc. Amer.*, vol. 36, pp. 2346–2353, 1964.

[39] T. Fujishima, "Realtime chord recognition of musical sound: A system using common lisp music," in *Proc. Int. Comput. Music Conf.*, Beijing, China, 1999, pp. 464–467.

[40] K. Lee and M. Slaney, "Acoustic chord transcription and key extraction from audio using key-dependent hmms trained on synthesized audio," *IEEE Trans. Audio, Speech, Lang. Process. Music Inf. Retrieval*, vol. 16, no. 2, pp. 291–301, 2008, Special Iss. Music Inf. Retrieval.

[41] E. Gómez, "Tonal description of music audio signals," Ph.D. dissertation, Univ. Pompeu Fabra, Barcelona, Spain, 2006.

[42] D. Ellis and G. Poliner, "Identifying cover songs with chroma features and dynamic programming beat tracking," in *Proc. Int. Conf. Acoust., Speech Signal Process.*, Honolulu, HI, 2007, vol. IV, pp. 1429–1432.

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

[44] M. Müller and S. Ewert, "Towards timbre-invariant audio features for harmony-based music," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 18, no. 3, pp. 649–662, Mar. 2010.

[45] A. Ockelford, *Repetition in Music: Theoretical and Metatheoretical Perspectives. Volume 13 of Royal Musical Association Monographs*. Farnham, U.K.: Ashgate, 2005.

[46] M. Slaney and M. Casey, "Locality sensitive hashing for finding nearest neighbors," *IEEE Signal. Process. Mag.*, vol. 25, no. 2, pp. 128–131, Feb. 2008.

[47] M. Li, X. Chen, X. Li, B. Ma, and P. Vitanyi, "The similarity metric," *IEEE Trans. Inf. Theory*, vol. 50, no. 12, pp. 3250–3264, Dec. 2004.

[48] R. Cilibrasi and P. M. B. Vitányi, "Clustering by compression," *IEEE Trans. Inf. Theory*, vol. 51, no. 4, pp. 1523–1545, Apr. 2005.

[49] J. Paulus, M. Müller, and A. Klapuri, "Audio-based music structure analysis," in *Proc. Int. Conf. Music Inf. Retrieval (ISMIR)*, Aug. 2010, pp. 625–636.

[50] D. A. Pelta, N. Krasnogor, C. Bousoño-Calzón, J. L. Verdegay, J. D. Hirst, and E. K. Burke, "A fuzzy sets based generalization of contact maps for the overlap of protein structures," *Fuzzy Sets Syst.*, vol. 152, no. 1, pp. 103–123, May 16, 2005.

[51] R. Middleton, , B. Horner and T. Swiss, Eds., "Form," in *Key Terms in Popular Music and Culture*. New York: Wiley-Blackwell, 1999, pp. 141–155.

[52] G. Widmer, S. Dixon, W. Goebl, E. Pampalk, and A. Tobudic, "In search of the Horowitz factor," *AI Mag.*, vol. 24, no. 3, pp. 111–130, 2003.

[53] W. J. Conover, *Practical Non-Parametric Statistics*, 3rd ed. New York: Wiley, 1998.

[54] M. Cebrián, M. Alfonseca, and A. Ortega, "Common pitfalls using the normalized compression distance: What to watch out for in a compressor," *Commun. Inf. Syst.*, vol. 5, no. 4, pp. 367–384, 2005.
[55] J. G. Cleary and I. H. Witten, "Data compression using adaptive coding and partial string matching," *IEEE Trans. Commun.*, vol. C-32, no. 4, pp. 396–402, Apr. 1984.
[56] F. Gouyon, A. Klapuri, S. Dixon, M. Alonso, G. Tzanetakis, C. Uhle, and P. Cano, "An experimental comparison of audio tempo induction algorithms," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 14, no. 5, pp. 1832–1844, Sep. 2006.
[57] J. P. Zbilut, A. Giuliani, and C. L. Webber, "Detecting deterministic signals in exceptionally noisy environments using cross-recurrence quantification," *Phys. Lett.*, vol. 246, no. 1–2, pp. 122–128, 1998.
[58] J. Rocha, F. Rosselló, and J. Segura, "Compression ratios based on the universal similarity metric still yield protein distances far from CATH distances," *Comput. Res. Repository (CoRR)*, vol. abs/qbio/0603007, Mar. 2006.

Juan P. Bello (M'06) received the Ph.D. degree in electronic engineering from Queen Mary University of London, London, U.K.

He was a Post-Doctoral Researcher and Technical Manager of the Centre for Digital Music, Queen Mary University of London. Since 2006, he has been an Assistant Professor of Music Technology at New York University, and a founding member of its Music and Audio Research Laboratory (MARL). He teaches and researches on the computer-based analysis of audio signals and its applications to music information retrieval, digital audio effects, and interactive music systems. He is a member of the IEEE and the society for music information retrieval (ISMIR), and a regular reviewer and contributor to digital signal processing and computer music journals and conferences. He is also a Researcher and member of the Scientific and Medical Advisory Board of Sourcetone, a music and health start-up.

Dr. Bello's work has been supported by scholarships and grants from Venezuela, the U.K., the E.U., and the U.S., including, more recently, a CAREER Award from the National Science Foundation.