A VIDEO COMPRESSION-BASED APPROACH TO MEASURE MUSIC STRUCTURE SIMILARITY

Diego F. Silva[1], Hélène Papadopoulos[2], Gustavo E.A.P.A. Batista[1] and Daniel P.W. Ellis[3]

[1]Instituto de Ciências Matemáticas e de Computação – Universidade de São Paulo
[2]Laboratoire des Signaux et Systèmes (L2S), CNRS UMR 8506, France.
[3]Department of Electrical Engineering - Columbia University

diegofsilva@icmc.usp.br, helene.papadopoulos@lss.supelec.fr, gbatista@icmc.usp.br, dpwe@columbia.edu

ABSTRACT

The choice of the distance measure between time-series representations can be decisive to achieve good classification results in many content-based information retrieval applications. In the field of Music Information Retrieval, two-dimensional representations of the music signal are ubiquitous. Such representations are useful to display patterns of evidence that are not clearly revealed directly in the time domain. Among these representations, self-similarity matrices have become common representations for visualizing the time structure of an audio signal. In the context of organizing recordings, recent work has shown that, given a collection of recordings, it is possible to to group performances of the same musical work based on the pairwise similarity between structural representations of the audio signal. In this work, we introduce the use of the Campana-Keogh distance, a video compression-based measure, to compare musical items based on their structure. Through extensive experiments, we show that the use of this distance measure outperforms the results of previous work using similar approaches but other distance measures. Along with quantitative results, detailed examples are provided to to illustrate the benefits of using the newly proposed distance measure.

1. INTRODUCTION

Within the last few years, time-series methods and algorithms have attracted the interest of many research communities such as Data Mining and Information Retrieval. Indeed, processes of interest generally change over time, and the study of how these changes occur is a central issue to fully understand such processes.

The choice of the distance measure between time-series representations can be decisive to achieve good classification results in many content-based information retrieval applications. Two distance measures commonly used in time-series analysis are the Euclidean distance (ED) and the Dynamic Time Warping distance (DTW). The DTW can be understood as an extension of the ED, able to provide nonlinear time scaling invariance, popularly known as warping [3]. Those simple and well-known distances have been successfully applied to various kinds of problems and have proven to be very hard to beat [9].

Some time-series features are not evident in the time domain. There is a large number of research papers that propose methods to explore alternative representations of time-series, such as autocorrelation [1] and shapelets [20], in order to clarify specific features.

In the hot context of organization and retrieval of large collections of music, the notion of similarity between music recordings is of great importance for many applications such as music summarization [8] or cover song retrieval [11]. In general, the similarity between two recordings is measured by comparing their respective time-series.

Music audio signals are highly structured at different time-scales (bar, phrase-level, sections, etc.) and exhibit repetitive segments, e.g. the so-called ABA sonata form (exposition - development - recapitulation) in classical music. Since their introduction in the domain of audio in 1999 [12], self-similarity matrices (SSMs) have become common representations for visualizing the time structure of an audio signal in terms of self-similarity and repetitions. Such two-dimensional representations are obtained by computing the pairwise similarity of an audio feature sequence (such as mel-frequency cepstral coefficients [21] or chromas [2]), and allow putting in evidence patterns that are not clearly revealed directly in the time domain. For instance, repeated patterns in the audio will appear as diagonal stripes in the SSM. For more details and a review about music structure, we refer the reader to [24].

Among the various applications based on cross-retrieval tasks, we are interested in this work in the problem of, given a piece of music as a query, to automatically retrieve from a given collection all performances (various interpretations) of the query. We focus on classical music. Two interpretations of the same piece will have a similar musical content, but they may differ in many ways. Besides articulation, phrasing and ornamentation, the global tempo may be different from one performance to another. Local tempo variations such as *ritardandi* may also exist. Moreover, other factors such as the recording conditions, the loudness or the instrumentation may result in huge dynamical and spectral deviations between the two interpretations.

Despite these variations, recent work has shown that it is possible to accurately measure the structural similarity between two recordings by computing the pairwise similarity between their self-similarity matrices, without extracting explicitly the underlying audio structure. In [16], this approach is applied to the cover song detection problem. In [22], this approach is used to build a retrieval system that searches a database for the musical piece that best matches a given *symbolic* structural query. Of particular interest to the present article are two closely related previous work that show that it is possible robustly group *audio* performances of the same musical work by using a distance that measures the pairwise similarity between their respective structural representations [4, 14].

In order to compare two-dimensional objects directly, it is appropriate to use a distance measure specific to this purpose. An example of that is the Campana-Keogh (CK-1) distance [7], which uses the video compression as the basis for estimating the dissimilarity between two images.

In this paper, we introduce the use of CK-1 to retrieve music recordings based on its structural similarity to an audio query. We show that the use of this distance measure outperforms the results of previous work using similar approaches but other measures.

2. STRUCTURAL SIMILARITY

As mentioned above, there are at least two studies that have successfully explored the possibility to retrieve music recordings according to their structural similarity to an audio query. In this work, we introduce the use of the CK-1 measure as an efficient distance that can be used to accomplish this task using self-similarity arrays.

This section describes each step of the proposed approach. We consider the following retrieval scenario. Given a query recording and a collection of music recordings that contains various performances of the same piece as the query, along with recordings of different compositions, we aim at retrieving all the performances of the query music piece. To this end, we define the training and retrieval steps by the Algorithm 1 and Algorithm 2, respectively.

Algorithm 1 Training phase

Require: v = A collection of music recordings
  $S \leftarrow []$
  for $i = 1 \rightarrow length(v)$ **do**
    $s \leftarrow SSM(v[i])$
    $S \leftarrow concatenate(S, s)$
  end for

2.1 Feature Extraction

The first step consists in extracting a set of features that provides relevant information about the musical structure. Among the various clues that humans use to determine the structure of a music piece, the harmonic progression

Algorithm 2 Retrieval phase

Require: q = A query recording
  S = The collection of SSM extracted in the training phase
  $s \leftarrow SSM(q)$
  $D \leftarrow []$
  for $i = 1 \rightarrow length(S)$ **do**
    $d \leftarrow d_{ck1}(s, S[i])$
    $D \leftarrow concatenate(D, \{d, i\})$
  end for
  $L \leftarrow sort\_by\_distance(D)$
  return L

is very important [6]. Since their introduction in 1999, the *Pitch Class Profiles* [13] or *chroma* features [25] became common features for describing the harmonic content of music. The chroma features are, in general, 12-dimensional vectors that represent the spectral energy of the pitch classes of the chromatic scale. They have been successfully used for various content-based retrieval tasks, especially in previous work on music similarity [4, 14].

Following these approaches, we extract a sequence of chroma features, as well as two variants of chroma-like features: the Chroma Energy Normalized Statistics (CENS) and the Chroma DCT-Reduced log Pitch (CRP). CENS features involve an additional temporal smoothing and downsampling step, leading to an increased robustness of the features to local tempo changes. CRP features boost the degree of timbre-invariance. We refer the reader to [23] for more details. For chromagram computation, we used the Matlab chroma toolbox [1]. All feature vectors are normalized to have unit norm.

2.2 Self-Similarity Matrix and Recurrence Plots

In order to analyze the music structure, we used a self-similarity matrix (SSM), defined by Equation 1.

$$S(i, j) = d(\vec{x}(i), \vec{x}(j)),\ i, j \in 1 \cdots N \tag{1}$$

where N is the length of the signal, $\vec{x}(i)$ and $\vec{x}(j)$ are the feature vectors at positions i and j of the signal, respectively, and $d(\cdot, \cdot)$ is a similarity measure. In this work, we used the cosine distance defined by Equation 2.

$$d(\vec{x}(i), \vec{x}(j)) = \frac{< \vec{x}(i), \vec{x}(j) >}{||\vec{x}(i)|| \cdot ||\vec{x}(j)||} \tag{2}$$

A well-known variation of the SSM is called Recurrence Plot (RP) [10], that introduces three parameters to the SSM: an embedding dimension m, a time delay τ and a closeness threshold ϵ. The general idea is that each vector $\vec{x}$ is augmented by m observations evenly spaced in τ units of time. Therefore, we end up with a matrix $X(i) \in \Re^{m \times k}$, composed by $\vec{x}(i)$, $\vec{x}(i+\tau)$, ..., $\vec{x}(i+(m-1)\tau)$. A RP can be formally defined according to Equation 3.

$$R_{i,j} = \Theta(\epsilon - d(X(i) - X(j))), X(i) \in \Re^{m \times k}, \quad i, j = 1..N - m \tag{3}$$

where ϵ is a closeness threshold and Θ is the Heaviside function (i.e. $\Theta(z) = 0$ if $z < 0$ and $\Theta(z) = 1$ otherwise). For a better understanding of these parameters and the use of closeness threshold, we recommend the reading of [15].

[1] http://www.mpi-inf.mpg.de/resources/MIR/chromatoolbox/

In short, RP is a thresholded version of the SSM, so that the SSM is transformed into a binary matrix. The general idea of using the parameter ϵ is to discard spurious differences among observations that may hinder the visualization of relevant events. However, we note that setting such a parameter is not an intuitive task, and one frequently has to rely on adhoc techniques to do so. As we will discuss in Section 5, our experiments show that the use of SSM and CK-1 distance can outperform the results obtained in the literature with RP and Normalized Compression Distance.

2.3 Compression Distances

The CK-1 distance [7] is based on the concept of Kolmogorov complexity, proposed to quantify the randomness of discrete sequences. The Kolmogorov complexity $K(x)$ of an object x is given by the size of the smallest program capable to output x on a universal computer, such as a Turing machine [19]. Intuitively, $K(x)$ is the minimal quantity of information required to generate a string x with a program. Similarly, the Kolmogorov conditional complexity $K(x|y)$ is the size of the smallest program to generate the sequence x, given a sequence y as auxiliary input.

These concepts are the basis to a metric called Information Distance, that is universal, in the sense that it subsumes other measures [5]. Although Information Distance has attractive theoretical properties, it is uncomputable in the general case. Therefore, several research papers have proposed approximations to this distance using compression algorithms [17, 18].

A widely used distance based on Kolmogorov complexity is the Normalized Compression Distance (NCD) [18], that uses standard compression algorithms to estimate the Kolmogorov complexity. Let $C(x)$ and $C(y)$ be the sizes of sequences x and y after they have been compressed, and $C(xy)$ the compression size of the concatenation of both sequences. The NCD is defined by Equation 4.

$$NCD(x,y) = \frac{C(xy) - \min\{C(x), C(y)\}}{\max\{C(x), C(y)\}} \quad (4)$$

NCD has shown to provide good similarity estimates for various applications in discrete domain, such as strings of DNA and natural language. NCD typically uses lossless compression algorithms, which are well-suited for discrete data. In short, lossless compression relies in finding *exact* recurring sequences in data. However, images are composed by real-valued pixels and exact repeating patterns are rare. Another relevant issue is that generally the used compression algorithms work over data sequences and are not able to make use of spatial patterns present in images.

To overcome these limitations, the CK-1 distance makes use of video compression algorithms. More specifically, it uses the MPEG-1 compressor. Such an algorithm explores recurring patterns within a frame and/or between consecutive frames to compress the video. When two consecutive frames are composed of similar images, the inter frame compression step should be able to exploit this structure to produce a smaller file size, which is therefore interpreted as significant similarity. As digital video is an important commercial application, many efforts have been made to achieve high compression rates in video encoding, making it a good approximation of the Kolmogorov conditional complexity.

In order to calculate a distance between two images, x and y, we can create a fictional video with one frame for each image. Thus, CK-1 is defined by Equation 5.

$$d_{ck1}(x,y) = \frac{C(x|y) + C(y|x)}{C(x|x) + C(y|y)} - 1 \quad (5)$$

where $C(x|y)$ is the size of a two-frame MPEG-1 video composed by images y and x, in this order.

3. RELATED WORK

In [4], the author proposes an approach to retrieve music using an *intermediate representation* of musical structure, without any explicit determination of such structure. The approach uses RP as representation of music structure and NCD as similarity measure between two plots. The paper performs a large-scale evaluation of the proposed approach, including an exploration carried out to find the best parameter configuration, a comparison of front-end features, embedding and recurrence analysis strategies. Supported by such a large experimental evaluation, the author provides evidence of the effectiveness of the approach.

Subsequently, a similar method was proposed in [14], using the Euclidean distance. Due to the simplicity of such distance, the proposed method resulted in a more efficient approach. However, temporal variations may cause recurrence shifts and the Euclidean distance is very sensitive to patterns translations. To overcome this limitation, the authors used a technique to smooth the resulting plot. The results, obtained on a dataset slightly different from [4] (with a smaller number of recordings) also proved effective.

4. EXPERIMENTAL SETUP

We conducted a broad experimental evaluation to assess the effectiveness of the proposed method for music retrieval. In order to evaluate and fairly compare our method to previous work, we used the same data sets, and the same feature extraction parameter variation used in [4]. In addition, we conducted experiments with a technique similar to that presented in [14], applying the binarization and blur methods over the SSM.

4.1 Data Sets

In our evaluation phase, we used two datasets of the classical repertoire, kindly provided by the author of [4]:

- The first dataset, referred in this paper as *123tracks*, is composed by 123 recordings of 19 different works from the classical and romantic period. Among these tracks, 56 are played on piano and the remaining 67 are symphonic movements. In total there are 59 different conductors in this dataset;

- The second dataset, referred in this paper as *Mazurkas*, consists in 2919 [2] recordings of 49 Chopin's mazurkas for piano. These were recorded by 135 different pianists.

4.2 Parameter Configuration

The CK-1 distance measure is parameter-free, as well as the procedure to create the SSMs. However, CK-1 has a small caveat, it requires that all images must have the same dimensions. Since the dimensions of the SSM are proportional to the recordings durations, which are variable, it is necessary to resize the feature sequences to a fixed dimension d before extracting the SSM. In our experiments, this is achieved by a resampling procedure, resulting in five different feature dimensions, $d \in \{300, 500, 700, 900, 1100\}$.

For each chroma-based feature, Chroma, CENS and CRP, we used seven different analysis window lengths, resulting in seven feature rates, $f \in \{0.333, 0.5, 1, 1.25, 2.5, 5, 10\}$ features/second.

To facilitate the reading and writing of parameter configurations, we adopted the notation $Feat_{f=F;d=D}$, where $Feat \in \{Chroma, CENS, CRP\}$ and F and D are the values of feature frequency (f) and the feature vector dimension (d), respectively.

4.3 Evaluation

All the experiments in this work were evaluated using mean average precision (MAP). Consider a collection C consisting of M items, and a subset $Q \subset C$ containing n different performances of the same music piece. Given a query $q_i \in Q$, we build a ranked list by arranging the results in ascending order according to the calculated distance between all the pieces in the collection and the query q_i. The Average Precision (AP) is defined by Equation 6.

$$AP(q_i) = \frac{1}{n} \sum_{r=1}^{M} P(r)\Omega(r) \tag{6}$$

where r is the rank in the ordered list, and

$$P(r) = \frac{1}{r} \sum_{i=1}^{r} \Omega(i) \tag{7}$$

and $\Omega(r)$ is 0 if the work r is relevant and 1 otherwise. Finally, the MAP is defined by the mean of all AP values.

5. RESULTS

5.1 Results on the *123tracks* dataset

We start presenting the results obtained with SSM and CK-1 distance on the *123tracks* dataset. Tables 1, 2 and 3 report MAP results obtained by the proposed method, varying parameters to calculate the SSMs using Chroma, CENS and CRP, respectively. Non-parametric paired Friedman and Dunnet post-hoc tests were applied to compare the statistical difference between the results. Yellow-shaded cells are statistically equivalent to the best result in each table.

Table 1. MAP results obtained with Chroma features

Feature rate (f)	Feature dimension (d) 300	500	700	900	1100
10	0.914	0.915	0.904	0.900	0.897
5	0.918	0.910	0.902	0.894	0.896
2.5	**0.930**	0.922	0.898	0.890	0.885
1.25	0.927	0.923	0.903	0.893	0.887
1	0.929	0.928	0.913	0.903	0.889
0.5	0.930	0.928	0.917	0.907	0.890
0.33	0.930	0.927	0.918	0.907	0.888

Table 2. MAP results obtained with CENS features

Feature rate (f)	Feature dimension (d) 300	500	700	900	1100
10	0.926	0.920	0.914	0.911	0.908
5	0.943	0.929	0.923	0.917	0.915
2.5	0.946	0.945	0.941	0.935	0.930
1.25	0.943	0.943	0.937	0.933	0.930
1	0.944	0.944	0.941	0.938	0.936
0.5	**0.946**	0.944	0.940	0.940	0.942
0.33	0.942	0.940	0.934	0.932	0.932

Table 3. MAP results obtained with CRP features

Feature rate (f)	Feature dimension (d) 300	500	700	900	1100
10	0.930	0.936	0.937	0.932	0.935
5	0.933	0.933	0.937	0.934	0.935
2.5	0.933	0.940	0.936	0.935	0.934
1.25	0.935	**0.941**	0.935	0.936	0.936
1	0.933	0.936	0.932	0.934	0.937
0.5	0.930	0.937	0.929	0.933	0.932
0.33	0.929	0.937	0.929	0,932	0.931

Since the parameter settings used in these experiments are the same as those used in [4], the results can be directly compared. The results obtained with the proposed method outperformed the results of [4] for all parameter settings. For instance, the best MAP result obtained with CK-1 and SSM is 0.946 ($CENS_{f=0.5;d=300}$). The competing method best result is 0.863 ($CRP_{f=10;d=700}$), a significant difference of almost 9%. We must note that CK-1 and SSM have no internal parameters to be searched for. Meanwhile, the recurrence plots used in [4] have three parameters that need to be set. According to the author, after a computational intensive search in the parameters space varying the closeness threshold, the time delay and the embedding dimension, the best result obtained was 0.921. We note that such result is still outperformed by our parameter-free method.

Although the performance difference is too small in order to claim a statistically significant difference with high confidence, there are other aspects that should be considered. The lack of internal parameters makes our method much simpler to use and having its results replicated by the research community. Another relevant issue is that our method is very robust to changes of *external* parameters. A poor parameter choice in the feature extraction

[2] This is the number of recordings presented in the official description of the dataset. However, there are pieces that are not available, and we found a total of 2914 recordings. This condition also applies to previous work, therefore does not affect the comparison of results.

step does not affect the performance significantly. For instance, the worst result obtained by the proposed method is 0.885, using $Chroma_{f=2.5;d=1100}$, while in [4], the worst result is 0.225. Such invariability to the parameter setting is verified by the lack of statistically significant differences among our best result and our remaining results when we vary the external parameters. We also obtained an excellent mean performance of 0.926 over all parameter values.

5.2 Further Experiments

We mentioned the fact that the NCD may not be appropriate to compare real value matrices. To prove this fact, we applied the NCD on the SSMs obtained with the best parameter configuration for Chroma, CENS and CRP, using the CK-1 distance. We also applied the NCD on the matrices obtained by setting $CRP_{f=10;d=700}$, the configuration that achieved the best result in [4]. The results of this experiment are shown in Table 4, as well as the results obtained using ED in the same configurations.

Table 4. Results obtained by applying NCD and ED on the SSM in some configurations.

Configuration	NCD	ED
$Chroma_{f=2.5;d=300}$	0.322	0.279
$CENS_{f=0.5;d=300}$	0.382	0.313
$CRP_{f=1.25;d=500}$	0.276	0.257
$CRP_{f=10;d=700}$	0.271	0.262

With this simple experiment, it is possible to note that ED and NCD are not appropriate to compare SSM directly. However, in [14], it was shown that the ED can be effectively used to retrieve songs by preprocessing the SSM. We take advantage of such idea to evaluate the use of CK-1 distance in this context.

As we did not have access to the code used in [14], we just simulated similar experiments by applying threshold and blur procedures. In other words, we did not apply path enhancement technique before applying the threshold. Our goal is not to directly compare the results, because we do not even have the same dataset. However, we can compare the use of ED and CK-1 when some preprocessing operations are applied on the SSMs.

In the binarization step (application of a threshold), we used the strategy to consider that $k\%$ of the closest points in the SSM represent recurrence. Thus, these points are transformed into black (0) pixels in the resulting RP. The remaining become white (1). To evaluate different scenarios, we used three different values for the threshold: $k \in \{10, 25, 50\}$. Furthermore, we used a two-dimensional circular averaging filter (Pillbox) to blur the image, using five different filter sizes: $l \in \{1, 5, 10, 20, 30\}$. Figure 1 shows different representations of the same recording of the Chopin's Prelude Op.28 No.4. Plot (a) represents the SSM, plot (b) represents the RP (using 25% of the points) and plots $(c$ & $d)$ the result of applying a blur filter on the RP with four different sizes.

For the sake of space, we do not present all the results of our experiments. Briefly, the best results obtained by the use of distance ED in this context were better than those obtained in [4], which used the RP-NCD, in most scenarios. However, the results obtained by CK-1 distance in the same context were better than ED in all scenarios. The best results obtained for each distance and each feature are presented in Table 5.

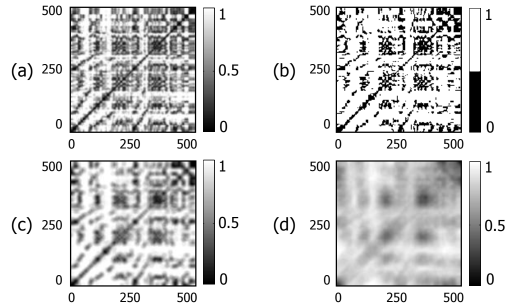

Figure 1. Four different representations of the same recording: (a) self-similarity matrix; (b) recurrence plot; (c) RP after application of a blur filter with size $l = 30$.

Table 5. Results obtained after applying a threshold and a blurring effect in the recurrence plots. The value k is the percentage of black points and l is the size of the blur filter. The symbol $*$ indicates where CK-1 is statistically outperformed ED in the same parameter configuration.

	k (%)	l	Distance	MAP
$Chroma_{f=2.5;d=300}$	50	1	CK-1	0.941$*$
			ED	0.816
	10	20	CK-1	0.905
			ED	0.872
$CENS_{f=0.5;d=300}$	25	5	CK-1	0.924$*$
			ED	0.829
	25	1	CK-1	0.919
			ED	0.910
$CRP_{f=1.25;d=500}$	25	30	CK-1	0.958$*$
			ED	0.893
	10	30	CK-1	0.953
			ED	0.941

5.3 Results on *Mazurkas* Dataset

We performed experiments on the *Mazurkas* dataset to validate the results obtained in the previous experiments. We first chose the parameter configuration $CENS_{f=0.5;d=300}$ since it obtained the best classification performance in the *123tracks* dataset. However, our method achieved a MAP of 0.611, which we considered unsatisfactory. A more detailed analysis of the SSMs showed that in many cases the matrices were not able to clearly represent the recording structure. This was due to the fact that the cosine distance, used to extract matrices, resulted in short distances in many cases. Thus, the figure generated by such distances contains very dark colors when applied to a color scale between 0 and 1.

After analyzing the recordings, we can conclude that the distances with small values can be directly related to the frequency in which the features were extracted. A low feature rate corresponds to a large analysis window, resulting in mixing several structurally distinct segments of music. Since many pieces in the dataset have a short duration and numerous structural variations of short duration, their structure can only be accurately analyzed with higher frequencies of feature extraction. To prove this fact, we performed the same experiment using $CENS_{f=1;d=300}$, achieving $MAP = 0.760$, without the need of normalizing the SSM, similar to the best result achieved in [4], $MAP = 0.767$. The Figure 2 shows an example of the difference between different feature extraction rates.

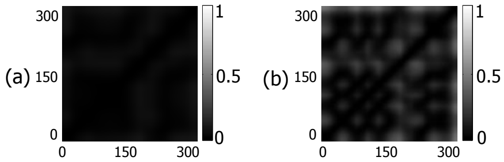

Figure 2. Self-similarity matrices of the same piece, but different feature rates: (a) $0.5f/s$; (b) $1f/s$;

Finally, we used the same dataset to test the configuration $CRP_{f=1.25;d=500}$ after the application of a threshold ($k = 25$) and a blur filter ($l = 30$). This test was performed since this configuration had the best result on the *123tracks* dataset. The effectiveness of the proposed method on this setting was proved in this experiment. When analyzing the structural similarity in this configuration using ED, we reach the result of $MAP = 0.652$. However, when we used the CK-1 measure, we obtained $MAP = 0.795$. This result is statistically superior to the results obtained by competing methods in the same dataset.

6. CONCLUSION

This paper proposes a simple and parameter-free approach to recover music by its structural similarity. The only parameters involved in our method are those required in the feature extraction step. We also show that our method is robust to a poor choice of these parameters, provided the parameter choices achieve meaningful SSM.

Through a wide experimental evaluation, we show that our method is superior to the techniques presented in previous work. Furthermore, CK-1 outperforms other distances regardless of pre-processing steps are performed in the signal or structural representation.

We believe that the contribution of this paper is not limited to the presentation of a new method for retrieving music by structural similarity. We hope this work will encourage the scientific community to analyze signals from various domains using visual representations and image-friendly distance measures.

7. ACKNOWLEDGMENTS

This work was supported by grants #2011/04054-2, #2012/-18985-0 and #2012/50714-7, São Paulo Research Foundation (FAPESP), and grant IIS-1117015, National Science Foundation (NSF).

8. REFERENCES

[1] A. Bagnall, L.M. Davis, J. Hills, and J. Lines. Transformation based ensembles for time series classification. In *ICDM*, 2012.

[2] M.A. Bartsch and G.H. Wakefield. Audio thumbnailing of popular music using chroma-based representations. *IEEE Trans. Multimedia*, 7(1):96–104, 2005.

[3] G.E.A.P.A. Batista, X. Wang, and E.J. Keogh. A complexity-invariant distance measure for time series. In *SDM*, 2011.

[4] J.P. Bello. Measuring structural similarity in music. *IEEE Trans. Sp. Aud. Proc.*, 19(7):2013–2025, 2011.

[5] C.H. Bennett, P. Gacs, Ming L., P.M.B. Vitanyi, and W.H. Zurek. Information distance. *IEEE Trans. Inf. Theory*, 44(4):1407–1423, 1998.

[6] M. J. Bruderer, M. F. McKinney, and A. Kohlrausch. The perception of structural boundaries in melody lines of western popular music. *Musicae Scientiae*, 13(2):273–313., 2009.

[7] B.J. L. Campana and E.J. Keogh. A compression based distance measure for texture. In *ICDM*, pages 850–861, 2010.

[8] M. Cooper and J. Foote. Automatic music summarization via similarity analysis. In *ISMIR*, 2002.

[9] H. Ding, G. Trajcevski, P. Scheuermann, X.e Wang, and E. Keogh. Querying and mining of time series data: experimental comparison of representations and distance measures. *VLDB*, 2008.

[10] J. P. Eckmann, Oliffson S. Kamphorst, and D. Ruelle. Recurrence plots of dynamical systems. *Europhysics Letters*, 4(9):973–977, 1987.

[11] D.P.W Ellis and G.E. Poliner. Identifying "cover songs" with chroma features and dynamic programming beat tracking. In *ICASSP*, volume 4, 2004.

[12] J. Foote. Visualizing music and audio using self-similarity. In *ACM Multimedia*, pages 77–80, Orlando, Florida,, November 1999.

[13] T. Fujishima. Real-time chord recognition of musical sound: a system using common lisp music. In *ICMC*, 1999.

[14] P. Grosche, J. Serr, M. Muller, and J.L. Arcos. Structure-based audio fingerprinting for music retrieval. In *ISMIR*, 2012.

[15] J.S. Iwanski and E. Bradley. Recurrence plots of experimental data: To embed or not to embed? *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 8(4):861–871, 1998.

[16] T. Izumitani and K. Kashino. A robust musical audio search method based on diagonal dynamic programming matching of self-similarity matrices. In *ISMIR*, 2008.

[17] E.J. Keogh, S. Lonardi, C.A. Ratanamahatana, L. Wei, S. Lee, and J. Handley. Compression-based data mining of sequential data. *Data Minining and Knowledge Discovery*, 14(1):99–129, 2007.

[18] M. Li, X. Chen, X. Li, B. Ma, and P.M.B. Vitanyi. The similarity metric. *IEEE Trans. Inf. Theory*, 50(12):3250–3264, 2004.

[19] M. Li and P. Vitanyi. *An introduction to Kolmogorov complexity and its applications*. Springer Verlag, second edition, 1997.

[20] J. Lines, L.M. Davis, J. Hills, and A. Bagnall. A shapelet transform for time series classification. In *SIGKDD*, 2012.

[21] B. Logan and A. Salomon. A music similarity function based on signal analysis. In *ICME*, 2001.

[22] B. Martin, M. Robine, and P. Hanna. Musical structure retrieval by aligning self-similarity matrices. In *ISMIR*, 2009.

[23] Meinard Müller and Sebastian Ewert. Chroma Toolbox: MATLAB implementations for extracting variants of chroma-based audio features. In *ISMIR*, Miami, USA, 2011.

[24] J. Paulus, M. Mueller, and A. Klapuri. Audio-based music structure analysis. In *ISMIR*, 2010.

[25] G.H. Wakefield. Mathematical representation of joint time-chroma distribution. In *ASPAAI*, 1999.