Signal Processing 117 (2015) 198–207

Contents lists available at ScienceDirect

Signal Processing

journal homepage: www.elsevier.com/locate/sigpro

Musical key extraction using diffusion maps

Ofir Lindenbaum [a,*], Arie Yeredor [a], Israel Cohen [b]

[a] Tel Aviv University, Tel Aviv 69975, Israel
[b] Technion-Israel Institute of Technology, Haifa 32000, Israel

ARTICLE INFO

Article history:
Received 8 December 2014
Received in revised form
30 April 2015
Accepted 10 May 2015
Available online 19 May 2015

Keywords:
Key extraction
Dimensionality reduction
Diffusion maps

ABSTRACT

We propose a method for automatic musical key extraction using a two-stage spectral dimensionality reduction (two consecutive mappings). First we build a data set representing the 24 Western musical keys, and then we use a nonlinear dimensionality reduction method, in order to understand the true manifold on which the musical keys lie. The order of the keys along the manifold is perfectly correlated with a cognitive model for the key space. We exploit this manifold in order to extract the musical key from a musical piece. Furthermore we propose three classifiers using the extracted manifold. The Classifiers work in two stages, by first estimating the mode and then by estimating the key within the estimated mode. Finally we examine our method on The Beatles data set and demonstrate its improved performance compared to various existing methods.

1. Introduction

Most Western musical pieces were written using the 24 possible diatonic musical keys. Musical keys describe the relation among pitches in the piece. This musical feature, which is retrievable by most musicians, is difficult to extract automatically using a computer. In this study we develop and examine an algorithm for automatic key extraction from raw data. Automatic key extraction has driven much recent research due to a large number of applications involved, for example, content based search [1], playlist generation, mosaicing, automatic accompaniment and disc jockey work. Recent studies focus on the task of extracting a musical key from raw data without the use of symbolic transcription; these studies have achieved reasonable results but cannot compete with a gifted musician.

To imitate the musician's intelligence, it is helpful to understand how an amateur musician might perform the task (at least according to the authors' own experience). By listening to a musical piece, the musician could create a tonal description (pitches or notes) of the song. Then, possibly by playing a musical instrument, the musician is able to determine which key is most appropriate for the given piece (a more proficient musician could improve this procedure). Most studies try to imitate this two-stage task by first extracting a tonal representation of the musical piece using a histogram of pitches [2] (12 semitones of the chromatic scale) represented in the chroma domain, or using a pitch class profile [3]. In the second stage the representation is classified into one of the 24 possible Western keys.

For the classification task Krumhansl and Kessler [4] have performed a cognitive probe-tone experiment and derived 24 typical key chroma profiles. These profiles aim to describe the significance of each pitch to the musical key; they represent the typical distribution of notes in a musical key. The profiles were modified by Temperley [5] and Gomez [6], who improved the classification results when analyzing Western music. The classification is performed by computing the correlation values to all 24

* Corresponding author. Tel.: +972 528783883.
E-mail address: ofirlin@gmail.com (O. Lindenbaum).

http://dx.doi.org/10.1016/j.sigpro.2015.05.006
0165-1684/

profiles and estimating the most prominent key. Recent algorithms such as [7] build a training set and derive a statistical average profile to represent each of the 24 keys. Both the statistical and the cognitive approaches for creating a typical profile can be described geometrically by a partition of the chroma space into 24 unrelated zones. However, the computation of the partition differs between the methods. In our study we will use a non-linear dimensionality reduction algorithm for the chroma space partition.

Many studies have been done in the field of cognitive music to model the structure of the tonal space [8]. One of the oldest models of this kind, which was developed by Leonard Euler in 1739 [9], is called Tonnetz (Tonal network), ordering all the harmonical similarities between notes in a 2-dimensional diagram. Other examples of these models are the Circle-of-Fifth, a double helix [10] and a torus [4]. In [11, 8] dimensionality reduction methods were used to visualize and confirm the existence of low dimensional structures in the tonal space. Chuan and Chew [12] used a Spiral Array space and proposed the Center of Effect model. According to this model each key is represented by a point on a spiral (more precisely a helix) and the classification is done by measuring the geodesic distance between points on the spiral. This approach can be viewed as creating an artificial manifold, then projecting a new data point to the manifold, and finally classifying the data using a partition of the manifold. Recent works by Peeters [13] use a Hidden Markov Model for the classification task.

The first hurdle is to extract the exact pitches from the musical piece. Due to the physical property of most musical instruments, playing a single note generates a fundamental frequency and all of its harmonics. This creates a problem when trying to extract pitches from a polyphonic musical piece because the harmonics of all of the instruments and human voices are mixed together. This problem can be solved by identifying the fundamental pitch and removing the harmonics from the spectral representation. These techniques have been used by Pauws [14]. Such models take into account the perceptual pitch and the musical background simultaneously. Chuan and Chew [12] proposed using a fuzzy analysis system, and Cremer and Derboven [15] proposed an overtone removal process. An alternative solution, implemented by Gomez [6], extends the Pitch Class Profile to Harmonic Pitch Class Profile by considering a theoretical amplitude contribution of the first four harmonics of each pitch within the three main triads in a given key. Genussov and Cohen [16] proposed approaching the problem using sparse representation methods. Various recent studies used Diffusion Maps (DM) [17,18], a non-linear dimensionality reduction method to extract and analyze unknown parameters from physical systems. These include among others: speaker identification [19], audio-visual recognition [20], classification of skeletal fibers [21]. In [8] tonal atonal classification was performed after applying DM to chroma representations of audio signal.

In this study, we extend this technique and address the challenging related task of key extraction. In the first part, we demonstrate the use of a dense DM for classification tasks of time varying signals. We show the improvement resulting from a dense time-domain blocks-processing, and we propose a novel approach for tuning the width-parameter of the DM kernel. The resulting dense diffusion mapping elucidates the low dimensional structure on which the keys lie (thereby corroborating the results of [11,8,22,23]). In the second part we use a two-stage mapping (one for the mode the other for the key) to propose three new classifiers of musical keys. Finally, we use the Beatles 179 songs data set as a test set and demonstrate the advantages of the proposed method compared to recent state-of-the-art algorithms.

The structure of the paper is as follows: Section 2 describes the methods and algorithms used and proposed in this work. In Section 3, we describe how we build and analyze the 24 keys training set. Experimental results are presented and analyzed in Section 4, followed by conclusions in Section 5.

2. Methods and algorithms

2.1. Tonal description

The first step of key extraction from raw audio data is extracting some tonal description of the musical piece. It is difficult to create an accurate transcription of the piece. However, we are not interested in a time representation of the piece, but rather in finding a description of the spectral energy corresponding to the pitches throughout the piece. We use a 12-D feature vector called Pitch Class Profile (PCP) to represent the tonal properties of the musical piece [2]. The PCP is derived from the chromatic scale. This scale is a 12 note musical scale, spaced with equal distances on a logarithmic scale starting at a basic note. It is a frequency domain vector showing the distribution of energy along the pitch classes [6] of a given musical piece. The frequencies are mapped onto a limited set of 12 chroma values (i.e., all octaves are wrapped into one). A common method for computing a PCP is the constant Q transform (CQT) [24] (used by [25] to track modulations in audio), a discrete spectral analysis of logarithmically spaced bins (similar to DFT). The L-bins CQT coefficients of a signal $\boldsymbol{s}[n]$ are computed as follows. The frequency range is first determined by selecting its lowest frequency $f_{\min}$ and its highest frequency $f_{\max}$. Then, denoting the desired number of bins per octave as β, the frequency center of the ℓ-th frequency bin set to $f_\ell = f_{\min} \cdot 2^{\ell/\beta}$, so that the total number of bins is $L = \beta \cdot \log_2(f_{\max}/f_{\min})$. The constant frequency-to-binwidth ratio is determined as $Q = (2^{1/\beta}-1)^{-1}$. The CQT coefficients are then given by

$$s_{\mathrm{cq}}[\ell] = \frac{1}{N_\ell} \sum_{n=o}^{N_\ell - 1} w_\ell[n] \cdot s[n] \cdot e^{-j2\pi n Q/N_\ell}, \quad \ell = 0,1,2,\ldots,L-1, \qquad (1)$$

where $w_\ell[n]$ is a window-function of n for extracting the ℓ-th CQT coefficient, and N_ℓ is the length of that window. The minimum required length of $w_\ell[n]$ is given (in samples) by $N_\ell = \lceil Q\frac{f_s}{f_\ell} \rceil$, where f_s is the sampling frequency.

Using $\boldsymbol{s}_{\mathrm{cq}} = [s_{\mathrm{cq}}[1], \ldots, s_{\mathrm{cq}}[72]]$, we compute the PCP vector $\boldsymbol{c}_s$ of $\boldsymbol{s}[n]$ by summing all corresponding bins from different octaves into a 12-D vector $\boldsymbol{c}_s$ whose bth element

is calculated by

$$c_s[b] = \sum_{p=0}^{P-1} |\mathbf{s}_{\mathrm{cq}}[b+p\beta]| \tag{2}$$

where b $(1 \le b \le 12)$ is the chroma bin number, and $P = \lfloor L/\beta \rfloor = 6$ is the total number of octaves in the CQT. In the current study, the PCP is further normalized to a maximal value of 1.

2.2. Dimensionality reduction

All dimensionality reduction algorithms aim at representing a given data set: $X = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_M\} \in \mathbb{R}^N$ by $Z = \{\boldsymbol{z}_1, \boldsymbol{z}_2, \ldots, \boldsymbol{z}_M\} \in \mathbb{R}^S$ such that $S \ll N$. Many methods exist for this task, e.g., Principal Component Analysis (PCA) [26] and Multidimensional Scaling (MDS) [27]. However, we assume that our data does not lie on a linear manifold, so we seek a method which at the same time preserves the local structure of the data. In our context the local structure is the connectivity between harmonically similar musical keys. The use of DM is well suited for this task as a nonlinear dimensionality reduction method. This method can help to model the relation between intrinsic latent parameters of musical pieces [28], such as Key, Timbre, Genre, Harmonic measures and more.

As described above, we have computed a 12-D representation of any given musical piece, residing in the nonnegative orthant of a 12-D space. However, this space is not fully occupied with musical data; the set of "legal" PCP vectors should be relatively sparse. This observation could be explained by the rules musicians obey to when playing music. For example, a uniformly spread PCP is highly unlikely to be extracted from a Western musical piece. Examining other similar examples will conclude in many other unoccupied points in the chroma space. We therefore seek a manifold which represents the "legal" chroma space of musical compositions. We use DM to extract only the meaningful dimensions from our chroma representation (PCP), and to find the manifold representing musical data within the 12-D space.

2.3. Diffusion maps

The DM framework is constructed by enforcing a Markov random walk model, based on the local connectivities of data points [8]. The random walk enables capturing the local relations within data points. Given a high dimensional data set $\boldsymbol{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3, \ldots, \boldsymbol{x}_M\} \subset \mathbb{R}^N$, the DM framework could be summarized by the following steps:

1. Choose a kernel function $\mathcal{K}: \boldsymbol{X} \times \boldsymbol{X} \longrightarrow \mathbb{R}$, represented by a matrix $\boldsymbol{K} \in \mathbb{R}^{M \times M}$ which satisfies for all $(\boldsymbol{x}_i, \boldsymbol{x}_j) \in \boldsymbol{X}$ the following properties: symmetry $K_{i,j} = \mathcal{K}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \mathcal{K}(\boldsymbol{x}_j, \boldsymbol{x}_i)$; positive semi-definiteness $\forall \boldsymbol{v_i} \in \mathbb{R}^M | \boldsymbol{v}_i^T \boldsymbol{K} \boldsymbol{v_i} \ge 0$; nonnegative values $\mathcal{K}(\boldsymbol{x}_i, \boldsymbol{x}_j) \ge 0$. *These properties guarantee real eigenvectors and nonnegative real eigenvalues of the matrix* $\boldsymbol{K}$. *A common example for such kernel is a Gaussian with an* L_2 *norm as the affinity measure between two data vectors*: $K_{i,j} = exp\{-\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2 / 2\sigma_x^2\}$.
2. By normalizing the kernel using $\boldsymbol{D}$: $D_{i,i} = \sum_j K_{ij}$, we compute the following matrix elements:
$$P_{i,j} = \mathcal{P}(\boldsymbol{x}_i, \boldsymbol{x}_j) = [\boldsymbol{D}^{-1}\boldsymbol{K}]_{i,j}. \tag{3}$$
The resulting matrix $\boldsymbol{P} \in \mathbb{R}^{M \times M}$ can be viewed as the transition kernel of a (fictitious) Markov chain on $\boldsymbol{X}$, such that the expression $[\boldsymbol{P}^t]_{i,j} = p_t(\boldsymbol{x}_i, \boldsymbol{x}_j)$ describes the probability of transition in t steps from point $\boldsymbol{x}_i$ to point $\boldsymbol{x}_j$.
3. Apply spectral decomposition to the matrix $\boldsymbol{P}$, or to one of its powers $\boldsymbol{P}^t$, to obtain a sequence of eigenvalues $\{\lambda_m\}$ and normalized eigenvectors $\{\psi_m\}$ satisfying $\boldsymbol{P}\psi_m = \lambda_m \psi_m, m = 0, \ldots, M-1$.
4. Define a new representation for the data set $\boldsymbol{X}$:
$$\boldsymbol{\Psi}_t(\boldsymbol{x}_i): \boldsymbol{x}_i \longrightarrow \left[\lambda_1^t \psi_1(i), \lambda_2^t \psi_2(i), \lambda_3^t \psi_3(i), \ldots, \lambda_{M-1}^t \psi_{M-1}(i)\right]^T \in \mathbb{R}^{M-1}, \tag{4}$$
where t is the selected number of steps and $\psi_m(i)$ denotes the ith element of $\boldsymbol{\psi}_m$.
The main idea behind this representation is that the Euclidian distance between two data points in the new representation is equal to the following weighted L_2 *distance between the conditional probabilities* $\boldsymbol{p}_t(\boldsymbol{x}_i, :)$ *and* $\boldsymbol{p}_t(\boldsymbol{x}_j, :)$ (*the i-th and j-th rows of* $\boldsymbol{P}^t$), *the following is usually called the Diffusion Distance* [17]:
$$\mathcal{D}_t^2(\boldsymbol{x}_i, \boldsymbol{x}_j) = \|\boldsymbol{\Psi}_t(\boldsymbol{x}_i) - \boldsymbol{\Psi}_t(\boldsymbol{x}_j)\|^2 = \sum_{m \ge 1} \lambda_m^{2t} (\psi_m(i) - \psi_m(j))^2 = \|\boldsymbol{p_t}(\boldsymbol{x_i}, :) - \boldsymbol{p_t}(\boldsymbol{x_j}, :)\|_{\boldsymbol{W}^{-1}}^2, \tag{5}$$
where $\boldsymbol{W}$ *is a diagonal matrix with elements*: $W_{i,i} = \phi_0(i) = D_{i,i} / \sum_{i=1}^M D_{i,i}$. *A proof of this equality can be found in* [17].
5. Choose a desired accuracy $\delta \ge 0$ for the diffusion distance defined above: $s(\delta, t) = \max\{\ell \in \mathbb{N}$ such that $|\lambda_\ell|^t > \delta |\lambda_1|^t\}$. Using the desired accuracy, define a new mapping of $s(\delta, t)$ dimensions
$$\boldsymbol{\Psi}_t^{(\delta)}: \boldsymbol{X} \to \left[\lambda_1^t \psi_1(i), \lambda_2^t \psi_2(i), \lambda_3^t \psi_3(i), \ldots, \lambda_s^t \psi_s(i)\right]^T \in \mathbb{R}^{s(\delta,t)}$$

Our choice of affinity measure in this work is an exponential kernel based on the cosine affinity between feature vectors. More specifically, denoting the 12-dimensional feature vector of the i-th musical frame as $\boldsymbol{x}_i$, let $\eta_i \triangleq (1/12) \sum_{b=1}^{12} x_i[b]$ denote its average value, and define the average-subtracted features as $\tilde{\boldsymbol{x}}_i \triangleq \boldsymbol{x}_i - \eta_i \cdot 1$ (where 1 denotes a 12×1 all-ones vector). The cosine affinity (which in our case is actually the correlation coefficient) between the i-th and j-th feature vectors is computed using

$$T_{ij} \triangleq \frac{\tilde{\boldsymbol{x}}_i^T \tilde{\boldsymbol{x}}_j}{\sqrt{(\tilde{\boldsymbol{x}}_i^T \tilde{\boldsymbol{x}}_i)(\tilde{\boldsymbol{x}}_j^T \tilde{\boldsymbol{x}}_j)}}, \quad i,j = 1, \ldots, M, \tag{6}$$

and the respective kernel-based distance measure is given by

$$K_{ij}(\sigma) = \kappa(\boldsymbol{x}_i, \boldsymbol{x}_j; \sigma) \triangleq \exp\left\{\frac{T_{ij} - 1}{2\sigma^2}\right\}, \quad i,j = 1, \ldots, M, \tag{7}$$

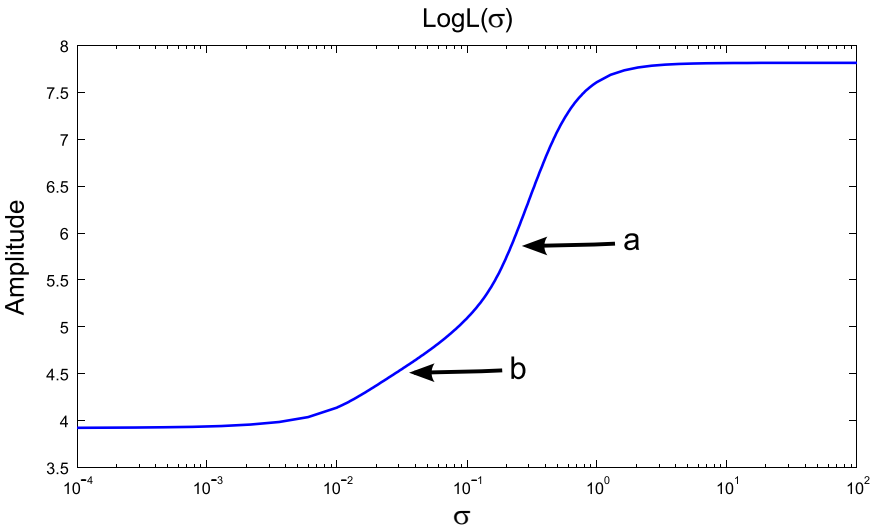

Fig. 1. The plot has two asymptotes. σ should be chosen in between the asymptotes, in the range where the plot appears linear. Two reasonable selections are indicated; selection "b" agrees with our alternative method.

where σ denotes an exponential-width parameter, which will be discussed in more detail in Section 2.5.

This distance measure neglects the influence of the chroma's energy. This suits our purpose as the chroma's energy has no impact on the musical key. Our main purpose is to classify the musical data, so we use the matrix $\boldsymbol{P}$ without running its powers ($t=1$); when the data lies continuously on a manifold (as in our case), running powers of $\boldsymbol{P}$ create more possibilities for connections on the graph of the data and make the classification task more difficult. The number of diffusion dimensions we use in this study varies from one classifier to another; however, the highest dimension used is $s(\delta,t)=12$, namely the full dimension of the features vector.

2.4. Artificial manifold

For effective use of DM, dense sampling of the data is required, so as to allow each data point to have a sufficient number of close neighbors, namely to lie on a dense manifold. In this work (as in many others in machine learning) we have collected a training set from independent sources. In general, such data does not lie on a dense manifold. For this reason we propose using overlapping time frames to artificially create a dense manifold. We chose to use successive frames of 30 s length with a 28-s overlap, each data point represents the average PCP vector of all the computed windows within the 30 s frame. The overlap guarantees that there is only a slight change in the PCP vector from point to point. In Section 3 we will present an extracted manifold constructed using our proposed overlapping scheme.

2.5. The choice of σ

The behavior of our chosen exponent kernel is determined by the width parameter σ in (7). A correct choice of σ preserves local connectivity and neglects the global distances. If σ is too large, there is almost no preference for the local connections and this method essentially reduces to PCA. On the other hand if σ is too small, the matrix K has many small off-diagonal elements, indicating poor connectivity within the data [8]. For this very common task in kernel-based methods several approaches have been proposed in the literature. Some choose σ as the standard deviation of the data, and this approach is good for capturing most of the data. However, we seek a value of σ which would be most effective for our classification task. We first consider a scheme proposed by Singer et al. [29]. Their scheme aims to find a range of values for σ. The idea is to compute the kernel at various values of σ and search for the values where the Gaussian bell shape exists. The proposed scheme can be implemented using the following 4 steps:

1. Compute: $\boldsymbol{K}(\sigma)$ for several values of σ.
2. Compute: $L(\sigma)=\sum_i\sum_j K_{ij}(\sigma)$ for these values.
3. Plot a logarithmic plot of $L(\sigma)$ (vs. σ).
4. Choose σ between the two asymptotes on a linear range of $L(\sigma)$.

Note that the two asymptotes would always be $L(\sigma)\xrightarrow{\sigma\to 0}\log(M)$, and $L(\sigma)\xrightarrow{\sigma\to\infty}\log(M^2)=2\log(M)$, since for $\sigma\to 0$, $\boldsymbol{K}$ approaches the Identity matrix, and for $\sigma\to\infty$, $\boldsymbol{K}$ approaches an all-ones matrix.

We applied this scheme to 8000 audio data points (the selection of data points will be explained in Section 3) and concluded a range of $\sigma\in[0.05\text{–}0.5]$. The plot demonstrating this process using our data is presented in Fig. 1. The plot has two asymptotes indicating two unwanted ranges in which the Gaussian kernel is not suited for the data.

In this work we propose an alternative scheme for a supervized selection of σ for classification tasks. Given a training set X and given I known classes $C_1, C_2, \ldots, C_I$, with G data points within each class (total data points: $J=G\cdot I$),

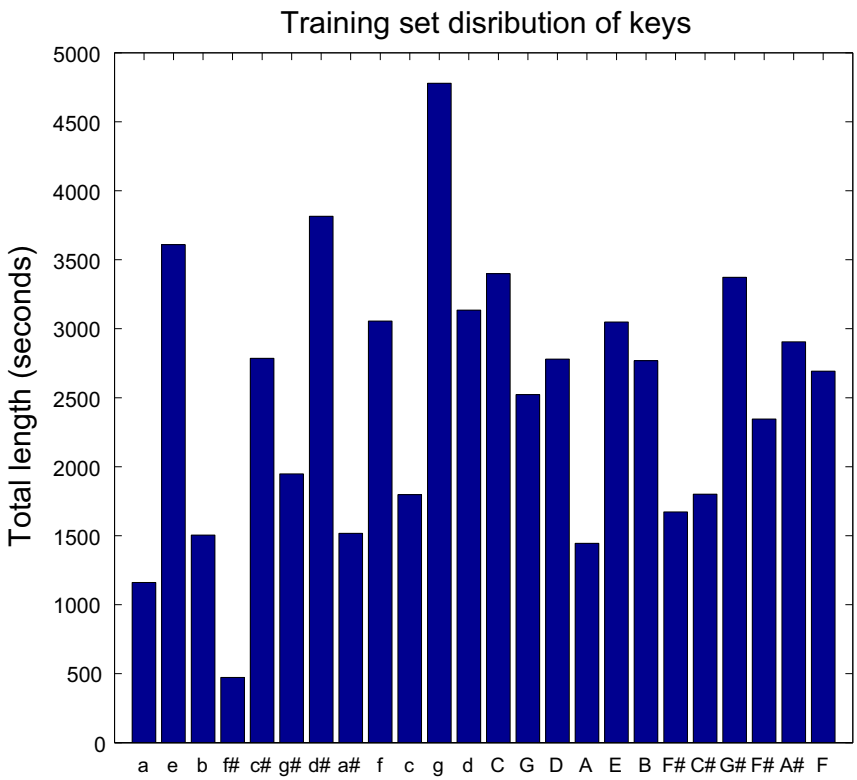

Fig. 2. Full training set (8000 pieces) consists of classical and improvization pieces total length used for each key (counting only the first and last minute of the classical pieces). Major keys – capital letter, Minor keys – minuscule letter.

compute several diffusion maps for various values of σ within the possible range. Using the $s(\delta)$ leading coordinates within each computed diffusion map $\Psi(\sigma)_{F\times(G\cdot I)}$, denote $\boldsymbol{\mu}_i$ as the center of mass of class C_i, computed in the diffusion coordinates, and μ_a as the center of mass of all data points. Compute the average square distance of the N data points from the center of mass within the class

$$D_{c_i} = \frac{1}{G}\sum_{\boldsymbol{x}_n \in C_i} \|\Psi(\boldsymbol{x}_n) - \boldsymbol{\mu}_i\|^2. \tag{8}$$

Then compute the same measure for all of the data:

$$D_a = \frac{1}{J}\sum_{\boldsymbol{x}_n \in X} \|\Psi(\boldsymbol{x}_n) - \boldsymbol{\mu}_a\|^2. \tag{9}$$

Finally, find σ which minimizes the ratio

$$\hat{\sigma} = \underset{\sigma}{\operatorname{argmin}} \frac{\sum_{i=1}^{I} D_{c_i}}{D_a}. \tag{10}$$

The idea is that this σ inherits the inner structure of the classes and neglects the mutual structure. Applying this scheme to our 8000 data points, we conclude a value of $\sigma = 0.078$ as a solution of (10). While the method proposed by Singer and Erban [29] provides a reasonable range for σ (as evident in Fig. 1), our method suggests a more specific, optimized value for σ. Nonetheless, our "optimal" value is still in good agreement with one of the reasonable values implied in Fig. 1.

2.6. Geometric harmonics

In order to extend the diffusion coordinates to a new data point (unlabeled musical piece) without re-applying a large-scale eigendecomposition, we use Geometric Harmonics [30]. We denote the training set, which was used to build the matrix $\boldsymbol{P}$, as X, and the rest of the new data set as Y (test set). The extended eigenvectors for a new data point $\boldsymbol{y} \in Y$ are approximated as weighted sums of the original eigenvectors, using our chosen kernel to compute the weights

$$\hat{\psi}_i(\boldsymbol{y}) = \frac{1}{\lambda_i}\sum_{\boldsymbol{x}_j \in X} \mathcal{P}(\boldsymbol{x}_j, \boldsymbol{y})\psi_i(j), \tag{11}$$

and the new mapping vector for data point

$$\hat{\Psi}(\boldsymbol{y}) = \left[\lambda_1\hat{\psi}_1(\boldsymbol{y}), \lambda_2\hat{\psi}_2(\boldsymbol{y}), \lambda_3\hat{\psi}_3(\boldsymbol{y}), \ldots, \lambda_{M-1}\hat{\psi}_{M-1}(\boldsymbol{y})\right] \in \mathbb{R}^{M-1} \tag{12}$$

The new coordinates in the diffusion space are only approximated and the new data points have no influence on the original map's structure.

3. Training and classification for the experimental results

In this study we used audio with a sampling rate of $f_s = 44.1$ KHz and chose $f_{\min} = 110$ Hz (A2) and $f_{\max} = 7040$ Hz (A8), so that with $\beta = 12$ bins per octave we get $Q \approx 16.8$, and a total number of $L = 72$ bins. We set

all the window sequences to standard rectangular windows, all of the same length of $N_\ell = 6364$, so that essentially the resulting coefficients are standard Discrete-Time Fourier Transform (DTFT) coefficients of a block of length N_1, taken on a logarithmic frequency-scale in the specified range. When processing audio segments of 30 s (1,323,000 samples) we accumulated the absolute values of the coefficients, extracted from all $\lfloor 1,323,000/6364 \rfloor = 207$ non-overlapping blocks.

We begin the key analysis by building a training set for all 24 possible Western keys. We downloaded 16.7 h of audio, out of which 35% were classical musical pieces by composers such as Bach, Pachelbel, Beethoven, Chopin, Wagner and many more. The classical pieces' keys were annotated by their composers. The other 65% of audio pieces are improvizations by amateur and professional musicians, all tagged according to the appropriate key along the improvization. These pieces include various instruments such as piano, organ, guitar, violin and even a quartet, and characterized by various genres such as rock, jazz, blues and pop. In Fig. 2 the lengths of the collected pieces are presented, indexed according to the tagged key.

For all pieces we ran a cross-annotation procedure; for each segment we computed the PCP and calculated the correlation value to the typical chroma profile [6] corresponding to the tagged key. If this value was negative we ignored the segment, if it was positive but lower than 0.5 we cross annotated it manually (by listening to the segment), otherwise we kept the segment and its original tag. We chose 8000 data points representing all 24 Western keys. Next, we applied random pruning so that the number of data points for each key is 200. This resulted in two training sets which consist of 2400 points each, one for the 12 major keys and the other for the 12 minors.

3.1. Low dimensional representation of keys

As explained in the first section, various low dimensional structures have been proposed to order the relations among keys and pitches. For example, 2-dimensional models such as a tonal network and a circle, or higher dimensional models such as a 4-dimensional torus, a helix and more (see, for example, [9,10,12,4]). The Circle-of-Fifths model was constructed according to the observation that shifting a key by a perfect fifth results in a harmonically similar key, meaning that both keys share notes and have many closely related harmonies.

Previous studies such as [11,8] have used dimensionality reduction schemes on pitch or key representations and found a correspondence between the extracted structure and Circle-of-Fifth model. To corroborate this correspondence and validate our training set, we applied DM to the training sets (both major and minor), using the optimal σ value that we computed according to the scheme explained in Section 2. Then we observed the leading dimensions of our manifold and compared them to the various geometrical models which result from cognitive musical studies. The two leading dimensions of the manifold for minor keys are presented in Fig. 3, and the mean of the classes for major keys can be observed in Fig. 4.

Using the two leading dimensions of our manifold we find a correlation to the Circle-of-Fifths, shape and the exact key order. This model does not appear when applying standard PCA on our training set. Furthermore, an examination of additional dimensions reveals a more complex manifold describing the tonal space.

3.2. Mode classifier

The 24 musical keys can be divided into two groups, 12 major and 12 minor keys. The major and minor attributes reflect the nature of a musical piece. For instance, the minor mode suggests for most Western listeners a negative emotional feeling [31]. Classifying the most prominent key from a musical piece is difficult due to the large number of possible classes. Using DM for this task is computationally expensive due to the large amount of data needed for the training set. We propose an initial classifier for the musical mode (i.e. major/minor) and a second classifier for the best key within the 12 possible keys. Both classifiers are based on DM, but use different training data and different dimensions. For the mode classifier we use 24 chroma profiles collected by Gomez [6]. We look for the diffusion coordinate which best describes the piece's mode, by looking for the coordinate having minimum variance inside the major/minor class and maximal variance between the classes; we do this by minimizing the ratio between these variances. We find that the 12-th diffusion coordinate (on Gomez' profiles), computed using a kernel as in Eq. (7), with $\sigma = 0.1$, best describes the mode parameter (in a manner that there is a binary separation between the modes). In Fig. 5, one can see two examples of the 12-th diffusion coordinate describing the 24 typical PCPs collected by Gomez [6] and our minor keys training set, respectively.

Major pieces fall closer to -1 and minor pieces fall closer to 1. However, many musical pieces combine the two attributes throughout the piece and tend to fall closer to 0. We use two thresholds for our mode estimator, $t_H = 0.5$ and $t_L = -0.5$. We denote the minimum distance to one of the 12 minor classes as $MinD_{\text{minor}}$ and the minimum distance to one of the 12 major classes as $MinD_{\text{major}}$. Given a test set $\boldsymbol{Y}$ and a representation $\hat{\boldsymbol{\Psi}}$ computed using (11) and (12), our proposed classifier could be implemented using Algorithm 1.

Algorithm 1. Mode classifier.

if $(\hat{\psi}_{12}(\boldsymbol{y}) \geq 0.5)$ **then**
  $\hat{y}_{\text{mode}} = \text{Minor}$
else
  if $(\hat{\psi}_{12}(\boldsymbol{y}) \leq -0.5)$ **then**
    $\hat{y}_{\text{mode}} = \text{Major}$
  else
    if $(MinD_{\text{major}} \geq MinD_{\text{minor}})$ **then**
    $\hat{y}_{\text{mode}} = \text{Minor}$
  else
    $\hat{y}_{\text{mode}} = \text{Major}$
    end if
  end if
end if

This classifier exploits two different DMs, one built from Gomez's profiles and the second built from our training

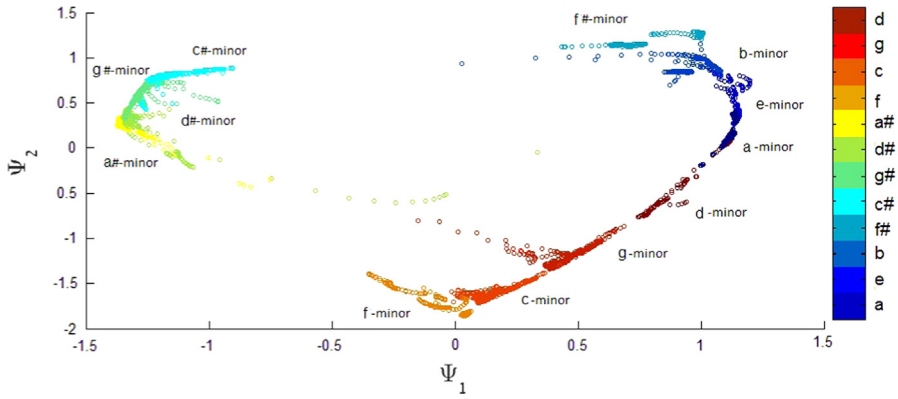

Fig. 3. Two dimensional representation of 2400 chroma vectors, sampled from pieces played in 12 minor keys. The Circle-of-Fifths order can be clearly observed.

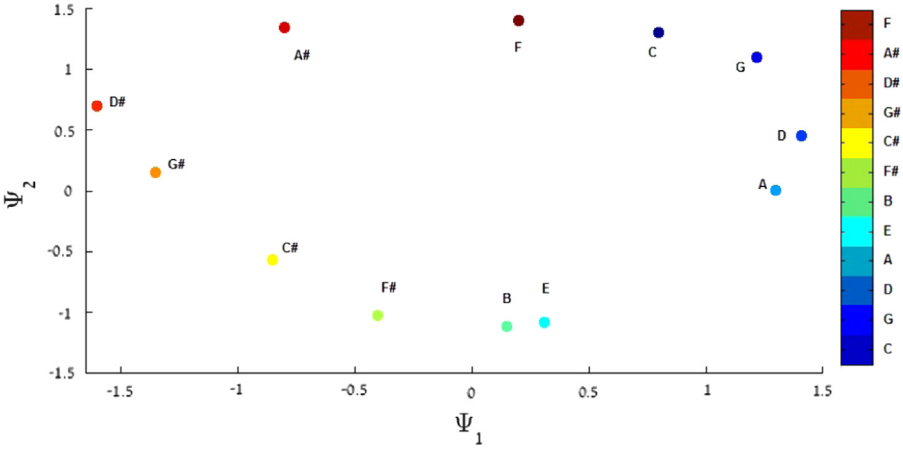

Fig. 4. The mean of the 12 major keys training set. The Circle-of-Fifths cognitive model appears when looking at a 2-D slice of the class' means.

set. The first DM is smaller by two orders, making the computational task much lighter.

3.3. Key classifier

To classify a new data point, we propose three classifiers. We denote the 4800 training points (major and minor together as explained in 3) and the corresponding diffusion map as X and Ψ. We use 72 chroma profiles collected by Krumhansl, Gomez and Temperley [32,6,5], we call them TypProfiles, and denote these profiles as $\overline{\boldsymbol{X}}$ and the corresponding mapping as $\overline{\boldsymbol{\Psi}}$. Fig. 6 demonstrates the classification power of DM on the 72 TypProfiles.

For all musical pieces in the test set $\boldsymbol{y} \in \boldsymbol{Y}$ we compute the diffusion coordinates in $\boldsymbol{\Psi}$ and in $\overline{\boldsymbol{\Psi}}$, using the Geometric Harmonics method (see Section 2.6). We classify each piece using the K-nearest neighbors method in both diffusion domains, using $s(\delta,1)=5$, since taking a higher dimension makes almost no change in the computed distances. The three different classifiers are denoted as:

Classifier I-COM (Center Of Mass - class mean):

1. Estimate the mode using Algorithm 1.
2. Find the nearest class mean in $\boldsymbol{\Psi}$.
3. Use the estimated mode and class to determine the key.

Classifier II-KNN (K-Nearest Neighbors):

1. Find the nearest training set points in $\boldsymbol{\Psi}$, use K-NN with K = 10.
2. Set the key as the majority vote within the 10 nearest neighbors.

Note that a KNN classifier based on a general training set can be quite sensitive to outliers. We therefore chose a relatively high value of $K=10$. This type of classifier is expected to be effective mostly when the tested pieces are typically similar (e.g., in genre or spectral properties) to the pieces in the training set.

Classifier III-TYP (Typical):

1. Estimate mode using Algorithm 1.
2. Find the nearest training set point in $\overline{\boldsymbol{\Psi}}$, use K-NN with $K=1$.
3. Use the estimated mode and class to determine the key.

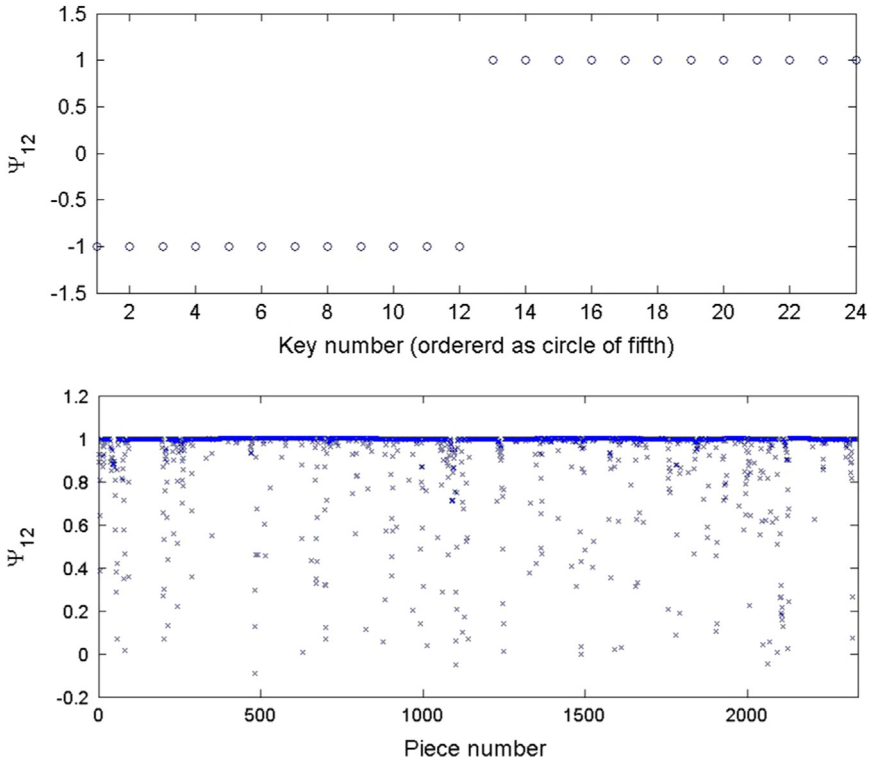

Fig. 5. Top – the 12th diffusion dimension of Gomez's profiles, indicating a binary separation between major and minor keys. Bottom – the 12th diffusion coordinate of 2400 Minor pieces, calculated using out of sample extension on Gomez' profiles. Average value of 0.946 with a variance of about 0.0274.

The TypProfiles Training set contains only 3 *samples within each class, therefore using* $K=1$ *and classifying helps us avoid conflicts when there are* 2 *or more nearest neighbors from different classes.*

4. Experimental results

For the experimental part we used a collection of 179 songs composed and performed by the Beatles. The keys of the pieces were taken from Pollack (1999) and cross annotated by a musicologist. Fig. 7 shows the distribution of keys used by the Beatles. This database was selected because it is easily available online and has been used for experiments in many previous studies. This database is very difficult to analyze due to the complex form of composition used by the Beatles, often with transition of keys within the songs, and because it contains percussive sounds and different postproduction effects.

To evaluate the algorithm properly and compare it to previous studies, we use a scoring method which is applied in the MIREX key finding competitions, as well as in other work such as [6,33]. According to this scoring system, performance is measured by the percentage of correctly identified keys as well as closely related keys. The idea is that some missanotated keys are more severe than others, because if the keys are related in some harmonic manner it can still be of some informative use to the musician; therefore we count a correct key as 1 point a perfect fifth (adjacent keys on the Circle-of-Fifths) as 0.5 points relative major/minor (same key wrong mode) as 0.3 points parallel major/minor (parallel keys on the Circle-of-Fifths) as 0.2 points.

Applying our algorithm to the first 30 s of the 179 musical pieces, with the parameters explained in this paper, we achieve (with the TYP classifier) an exact classification rate of the key in 66.5% and an average score of 75.6%. In Table 1, we present a summary of previous methods attempting key classification on this database, and majority of the results were simulated by Gomez [6]. We note that within this data set there are ambiguities regarding a few songs, nonetheless, we have not discarded these songs, so as to maintain a fair comparison to Gomez' results.

We note that there are two more works by Rocher et al. [34] and Papadopoulos and Tzanetakis [35], applying their key extraction methods to the same database; however, they use only 174 and 141 songs, respectively, within this database, discarding particularly difficult or ambiguous pieces, therefore a comparison to their results might be somewhat misleading. Their exact classification rate and average score are 62.4% and 82.27%, respectively.

The methods in Table 1 are ordered by descending scores and are annotated by the names of their respective authors. Each method was examined using two additional variations, which are a pre-processing stage for the chroma vector: in one variation only the three most dominant notes are taken into account; in the other the three dominant notes have been weighted according

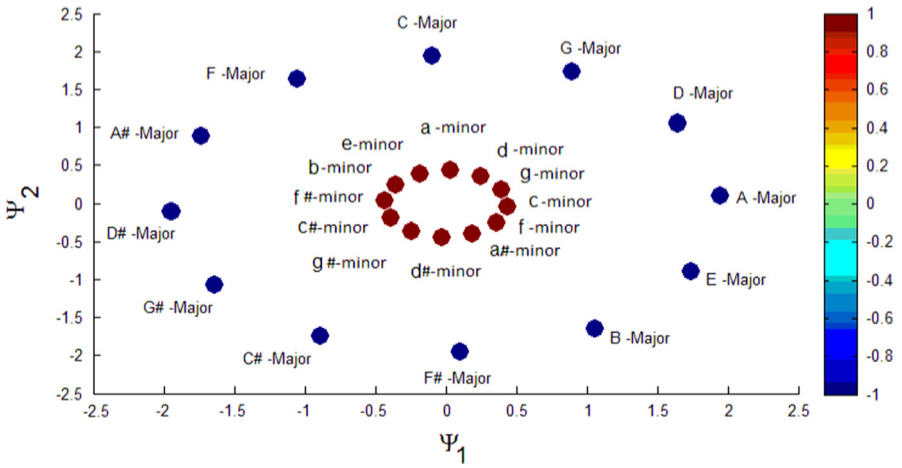

Fig. 6. The two leading dimensions of the 72 "typical" profiles, colored by the value of the 12th coordinate. Every point represents 3 profiles of the same key. The exact shape and order of Circle-of-Fifths is again evident.

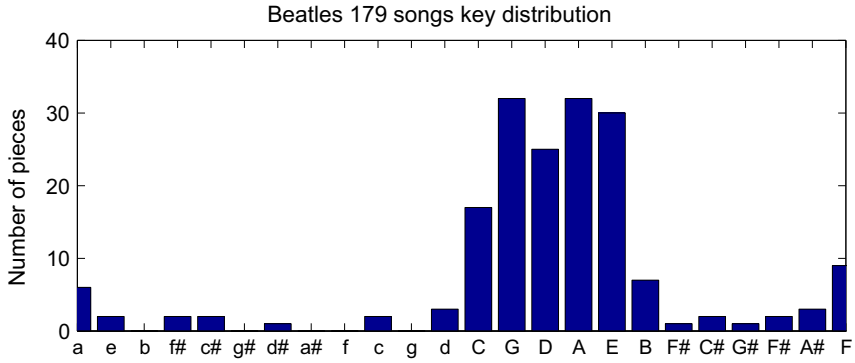

Fig. 7. Distribution of keys within the 179 Beatles songs database.

Table 1
Summary of results of key extraction on the Beatles database as simulated by Gomez, compared to our 3 proposed methods.

Method	Score (%)	Exact (%)	Mode (%)
TYP	**75.6**	**66.5**	**92.18**
Temperley-E-1	74.17	66.29	88.57
COM	**73.96**	**64.25**	**92.18**
Temperley-2	73.2	63.43	87.43
Gomez-1	72.91	65.14	81.14
Krumhansl–2	72.46	61.71	86.86
KNN	**71.46**	**61.46**	**86.59**
Krumhansl–1	71.37	60	86.29
Temperley-1	70.69	62.86	90.29
Krumhansl–3	69.43	63.43	85.71
Triad-1	68.57	60.57	78.29
Temperley-E-2	68.4	54.86	85.71
Gomez-2	64.4	46.86	85.71
Temperley-3	63.89	57.14	84
Chai-1	62.74	54.29	71.43
Temperley-E-3	59.89	45.14	81.71
Chail-2	57.2	40.57	73.71
Gomez-3	53.43	41.71	71.43
Chail-3	51.2	35.43	70.29
Triad-2	48.91	20.57	84.57
Triad-3	45.49	18.86	83.43

to their appearance in a chord. The exact estimation, Mirex-score, and mode percentage are presented in the corresponding columns. Evidently, we improve the results in all three measures for key classification. As can be concluded from the results presented in Table 1, the mode estimation procedure used for the COM and TYP classifiers yields most of the classification improvement compared to the results of state of the art method by Temperley and Gomez. The training set gave us insight regarding the geometric model representing the keys; however, the resulting KNN performance is only slight and is inferior to Temperley and Krumhansl's modified methods and offers only marginal improvement over the other methods. The distribution of the exact and related keys classification is presented in Fig. 8, we have chosen to present the leading scores from each method.

5. Conclusions and future work

We have shown that adapting DM to musical data reveals musical features (such as key or mode) and helps extract meaningful low dimensional mappings of these features. The method is very effective when the training set and the test set

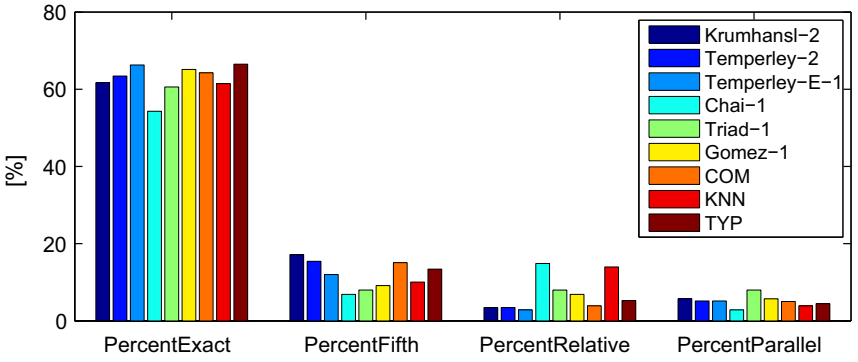

Fig. 8. Distribution of scores for the Beatles database, 6 leading methods.

have similar features, for instance, belong to the same genre or have the same timbre. In order to use this method on a "blind" test set (unknown musical properties), a training set containing various genres of music must be built. We have presented an automated algorithm for building a diffusion map, which is best suited for classification tasks on any type of data. However, this paper is not meant to focus on the classification task per se, but rather to demonstrate the use and the insights obtained from the DM in the context of Key extraction, comparing to some other competing classification approaches. Future work should involve examining the proposed a two-stage DM and σ selection method on other data sets. It would be interesting to examine this method for extracting other musical features such as pitch, timbre and chord, as they are not perfectly retrievable using current studies.

References

[1] O. Lindenbaum, S. Maskit, O. Kutiel, G. Nave, Musical Features Extraction for Audio-based Search, IEEEI, Eilat, 2010.
[2] M. Bartsch, G. Wakefield, To catch a chorus: using chromabased representations for audio thumbnailing, in: Proceedings of IEEE Workshop on Application of Signal Processing to Audio and Acoustics, WASPAA, 2001, pp. 15–18.
[3] T. Fujishima, Realtime chord recognition of musical sound: a system using common lisp music, in: Proceedings of International Computer Music Conference, ICMC, 1999, pp. 464–467.
[4] C. Krumhansl, E. Kessler, Tracing the dynamic changes in perceived tonal organization in a spatial representation of musical keys, Psychol. Rev. 89 (4) (1982) 334–368.
[5] D. Temperley, What's key for key? The krumhansl-schmuckler key-finding algorithm reconsidered, Music Percept. 17 (1) (1999) 65–100.
[6] E. Gomez, Tonal description of polyphonic audio for music content processing, INFORMS J. Comput. 18 (3) (2006) 294–304.
[7] S. van de Par, S. McKinney, A. Redert, Musical key extraction from audio using profile training, in: Proceedings of ISMIR, 2006, pp. 328–329.
[8] Ö. İzmirli, Tonal-atonal classification of music audio using diffusion maps, in: 10th International Society for Music Information Retrieval Conference, 2009.
[9] L. Euler, Tentamen novae theoriae musicae ex certissismis harmoniae principiis dilucide expositae, Saint Petersburg Academy, Saint Petersburg, 1739, p. 147.
[10] R.N. Shepard, Geometrical approximations to the structure of musical pitch, Psychol. Rev. 89 (4) (1982) 305–333.
[11] J.A. Burgoyne, L.K. Saul, Visualization of low dimensional structure in tonal pitch space, in: ICMC 2005, 2005.
[12] C. Chuan, E. Chew, Polyphonic key finding using the spiral array ceg algorithm, in: Proceedings of the International Conference on Multimedia and Expo, vol. 3, 2005, pp. 21–24.
[13] G. Peeters, Musical key estimation of audio signal based on hidden markov modeling of chroma vectors, in: Proceedings of 9th International Conference on Digital Audio Effects, 2006, pp. 127–131.
[14] S. Pauws, Musical key extraction from audio, in: Proceedings of 5th International Conference on Music Information Retrieval, 2005, pp. 96–99.
[15] M. Cremer, C. Derboven, A system for harmonic analysis of polyphonic music, in: AES 25th International Conference, 2004, pp. 115–120.
[16] M. Genussov, I. Cohen, Multiple fundamental frequency estimation based on sparse representations in a structured dictionary, Digit. Signal Process. 23 (1) (2013) 390–400.
[17] R.R. Coifman, S. Lafon, Diffusion maps, Appl. Comput. Harmon. Anal. 21 (2006) 5–30.
[18] S. Lafon, Y. Keller, R.R. Coifman, Data fusion and multicue data matching by diffusion maps, IEEE Trans. Pattern Anal. Mach. Intell. 28 (11) (2006) 1784–1797.
[19] Y. Michalevsky, R. Talmon, I. Cohen, Speaker identification using diffusion maps, in: 19th European Signal Processing Conference, 2011.
[20] Y. Keller, R. Coifman, S. Lafon, S. Zucker, Audio-visual group recognition using diffusion maps, IEEE Trans. Signal Process. 58 (1) (2010) 403–413.
[21] R. Neji, G. Langs, J.-F. Deux, M. Maatouk, A. Rahmouni, G. Bassez, G. Fleury, N. Paragios, Unsupervised classification of skeletal fibers using diffusion maps, in: IEEE International Symposium on Biomedical Imaging, 2009, pp. 410–413.
[22] Ö. İzmirli, Estimating the tonalness of transpositional type pitch-class sets using learned tonal key spaces, in: Mathematics and Computation in Music, Springer, 2009, pp. 146–153.
[23] G.K. Sell, Diffusion-based music analysis, (Ph.D. dissertation), Stanford, 2010.
[24] J. Brown, Calculation of a constant q spectral transform, Acoust. Soc. Am. 89 (1) (1991) 425–434.
[25] B. Blankertz, K. Obermayer, H. Purwins, Constant q profiles for tracking modulations in audio data, in: International Computer Music Conference, 2001, pp. 407–410.
[26] I. Jolliffe, Principal Component Analysis, vol. 21, 2005.
[27] J.B. Kruskal, W.M. Multidimensional Scaling, Sage Publications, Beverly Hills, 1977.
[28] R. Talmon, D. Kushnir, R. Coifman, I. Cohen, S. Gannot, Parametrization of linear systems using diffusion kernels, IEEE Trans. Signal Process. 60 (3) (2012) 1159–1173.
[29] A. Singer, R. Erban, I.G. Kevrekidisc, R.R. Coifmand, Detecting intrinsic slow variables in stochastic dynamical systems by anisotropic diffusion maps, 106 (38) (2009) 16090–16095.
[30] R.R. Coifman, S. Lafon, For what filters is every reduced product saturated? Appl. Comput. Harmon. Anal. 21 (2006) 31–52.
[31] M.P. Kastner, R.G. Crowder, Perception of the major/minor distinction: Iv. emotional connotations in young children, Music Percept. (1990) 189–201.
[32] C. Krumhansl, Cognitive Foundations of Musical Pitch, vol. 17, Oxford University Press, 1990, 307.
[33] Ö. İzmirli, Audio key finding using low-dimensional spaces, in: ISMIR, 2006, pp. 127–132.
[34] T. Rocher, M. Robine, P. Hanna, L. Oudre, Concurrent estimation of chords and keys from audio, in: Proceedings of the 11th International Society for Music Information Retrieval Conference, August 9–13 2010, pp. 141–146.
[35] H. Papadopoulos, G. Tzanetakis, Modeling chord and key structure with markov logic, in: 13th International Society for Music Information Retrieval Conference, 2012.