Musical Scales Recognition via Deterministic Walk in a Graph

Andrés Eduardo Coca S.
COENC - Coordinação de Engenharia em Computação
Universidade Tecnológica Federal do Paraná (UTFPR)
Toledo, PR – Brasil
E-mail: andressalazar@utfpr.edu.br

Liang Zhao
DCM - Departamento de Computação e Matemática
Universidade de São Paulo (USP)
Ribeirão Preto, SP – Brasil
E-mail: zhao@usp.br

***Abstract*—Musical scales play an important role in melodies, since its properties are reflected to the melodic essence. The extraction and understanding of scales are essential in both analysis and composition of music. However, the scale identification is a nontrivial task. Consequently, classic algorithms for identifying scales have been developed based on the most popular scales, such as major and minor scales. In this paper, we propose a comprehensive method for identifying musical scales, which allows to detect a wide range of scales beyond the traditional ones. Our method uses a deterministic walk through the nodes of a graph, where each node represents a valid interval structure. The transition between nodes is performed following a validation rule that governs the fragmentation of intervals. Moreover, if the scale is incomplete, possible structures can be determined and the scale is estimated according to the harmonic similarity percentage measure. The proposed method has been tested using a database of Finnish folk melodies and a data set of random melodies composed using rarely used scales. Experimental results show good performance of the proposed technique.**

Keywords*—*musical knowledge extraction; graphs; musical scales; codification; deterministic walks.

I. Introduction

The properties and acoustic characteristics of a musical scale are reflected in the melody and contribute with the essence of musical genre. So the scales play a fundamental role within the music and its study and analysis is of great interest in musical composition theory [17] and in several areas of research as Music Information Retrieval (MIR). For example, some studies found in MIR are the characterization of the scales used in traditional Thai music [11] and evaluation of happiness and embarrassment of the major and minor modes [20], among others.

Scale identification of a melody is necessary in many large musical projects, but it is still a challenge task. Consequently, in the literature it has been proposed different approaches. The first algorithm for detecting the tonality to be implemented in a computer was the Longuet-Higgins and Steedman's algorithm [14]. This algorithm compares the tones of the input musical work with the tonal region of each of the major and minor keys. This idea has motivated the development of the well-known algorithm for tonality detection in tonal music, the Krumhansl-Schmuckler's algorithm [4]. This algorithm correlates the distribution of pitch-class weighted according to the durations with the 24 profiles of the major and minor scales studied by Krumhansl and Kessler in [5]. From then on, other algorithms have been developed based on this idea [8] [16].

As seen, some approaches need prior information about the profile of tonalities to be detected, and these are, generally, the major/minor scales. However, different studies about scales enumeration have shown that the total number of scales is great [10], although in Western music is only used one small part [15]. Thus, there are many possibilities to select one or more scales for use in a composition and then identifying the musical scale is not limited to decide between major/minor scale but to determine the scale used across a wide range of options. Moreover, many melodies and musical genres use different scales to traditional, for example jazz and blues [17]. Thus, the development of an algorithm to identify the musical scale of a melody composed by using a different scale to the traditional is required.

In this paper we propose an algorithm to identify the musical scale of a monophonic melodic without modulation, which is composed of any intervallic structure and not necessarily the structure of the most popular scales. The proposed method does not need prior knowledge of the profile scale and works for any scale of the Twelve-tone Equal Temperament (12-TET) with structure intervals of 1, 2 and 3 semitones. Specifically, the algorithm can detect 11124 different scales.

On the other hand, most scales do not have a defined name and the names are not standardized, e.g., the harmonic minor scale is also known as Mohammedan scale [17]. Musical coding systems are very useful in representation, composition and musical knowledge extraction, for example, extraction of rhythmic patterns [2], melodic contours notation [7] and musical cryptograms [18], among others. Thereby, this paper also proposes a scale encoding system, where each scale is identified in a unique way with a numeric vector. Thus, the proposed method identifies and returns the scale code but not its name (if any).

The scale detector performs a deterministic walk through the nodes of a predefined graph. In this graph each node is an interval structure and the edges represent the possible transformations that may have a intervallic structure when its intervals are fractionated. The walk between nodes is determined by a validation rule, which determines whether adding a new interval corresponds to a correct structure.

The proposed method was tested for melodies composed of both popular and seldom used scales. In the first case, we used

TABLE I. TOTAL NUMBER OF SCALES WITH DIFFERENT STRUCTURES (T_E), TOTAL MODES (T_M) AND TOTAL SCALES TRANSPOSED (T_{ET}) IN N-TET SYSTEM FOR DIFFERENT VALUES OF N (APPROXIMATE VALUES IN SCIENTIFIC NOTATION).

N	T_E	T_M	T_{ET}
12	351	2048	24576
24	699251	8.3×10^6	201.3×10^6
36	1.9×10^9	34.3×10^9	1.2×10^{12}
48	5.8×10^{12}	140.7×10^{12}	6.7×10^{15}

the database of Finnish folk music [19], and for the latter, we created a database of random melodies and using scales with different intervallic structures. In each case, if not used all the notes of the scale, the algorithm determines the possible intervallic structures and estimates the most appropriate scale according to the harmonic similarity percentage measure. This measure relates the preferably of a scale with its spectral characteristics [12].

This paper is organized as follows: Section II describes the method of identifying musical scales step by step. The Section III presents the experimental results and Section IV presents the conclusions.

II. THE PROPOSED METHOD FOR IDENTIFYING MUSICAL SCALES

The proposed method for detecting the musical scale of a melody consists of the following steps: pitch selection in symbolic data, representation and encoding of musical scales, deterministic walk through the intervallic structure digraph, scale code calculation of the last node visited.

A. Representation and encoding of musical scales

In music theory scales are identified with a textual name, which is only defined for the most common scales, it does not provide all the necessary information to build the scale and difficult the standardization. Consequently, in literature it is possible to find systems to catalog scales, since knowledge and sorting them facilitates its handling in some specific task. In [6] was proposed a method of encoding scales that is used in the analysis of a set of notes, and in music theory is used the numerical nomenclature called Forte number [3].

In tasks that require the availability of a large number of scales, it can be used a standardized table with the name of the scale and some type of representation showing how to build it, but in the N-TET system to create a table of this type is not feasible because the number of different scales is large and grows exponentially as N [15]. For example, for $N = 12$ the total number of scales with different structures (T_E) is 351 [10], that altogether sum 2048 modes (T_M), that when transported to the 12 notes of the chromatic scale generate 24576 scales (T_{ET}) [15]. In case of $N = 12$, create a table with 351 entries is an acceptable size, and therefore viable, but for microtonal systems ($N > 12$) becomes unattainable. Table I shows the value of T_E, T_M and T_{ET} for different values of N. To avoid using a table of this type we propose in this section a numerical code for identification and construction of scales.

A scale can be represented by a interval vector, whose elements indicate the interval semitones between successive notes that conform the scale, e.g., the interval vector of the major scale is [2 2 1 2 2 2 1]. The scale can be fully constructed by using the interval vector, mode and tonic. Therefore, for generating all scales for a given value of N is necessary to generate all possible vectors intervals. This can be performed using a special combinational algorithm [9]. A common property of combinatorial algorithms is the preservation of lexicographical order in the organization of the generated objects. The lexicographical order for a finite set is the organization of the elements similar to the alphabetical order of words in a dictionary. Due to the above, the lexicographical order is preserved in the proposed code.

First, in this paper, the scales are classified into three classes, namely: primary (also known as type of scale [15], no equivalent scale or megamode [6]), secondary and tertiary. The primary scale is the one created from a circular permutation in lexicographical order. Secondary scales are derived from the circular rotation of a primary scale, that is, the secondary scale is a mode of a primary scale. Tertiary scale comes from the transposition of a primary or a secondary scale.

For the 12-TET system we encode scales with intervals of 1, 2 and 3 semitones[1] known as semitone, whole tone and tone and a half tone, respectively. However, scales with intervals between between 4 and 6 semitones are also possible [15].

Altogether, the system allows encoding 132 primary scales, 927 secondary scales and 11124 tertiary scales, as shown in Table III. The total number of intervals of each type (S, T or T_m) forming the architecture of the musical scale is stored in the set $u = \{s, t, t_m\}$, called scale structure [1]. Note that S represents the semitone interval and s indicates the total number of semitone. The pairs T and t and T_m and t_m are defined in a similar way.

The scale can be completely encoded using 6 values, which are arranged in a vector $\mathbf{c}$, called the scale code vector, so: $\mathbf{c} = [\ t_m \quad n \quad g \quad \eta \quad m \quad \tau\ ]$. Each of these elements are described below:

1) t_m: It indicates the number of tone and a half tone of the scale structure, and may have values in the range $0 \leq t_m \leq 4$.
2) n: It is the number of notes of the scale and may have values in the range $4 \leq n \leq 12$.
3) g: In some cases, a scale structure may have primary scales with different quantity of modes. The primary scales are grouped according to the total number of possible modes (see Table III). The third value g represents the group of the primary scale and may have values in the range $1 \leq g \leq 3$, where the first group has the largest number of modes.
4) η: For each group g, the primary scales are numbered in ascending order according to its lexicographical position within the total set. This position is represented in the code with $\eta \in [1, 21]$ and may have values in the range $1 \leq \eta \leq 21$.
5) m: It indicates the mode of the scale and may have values in the range $1 \leq m \leq 11$.
6) τ: The last value indicates the scale's tonic and has values in the ranges $0 \leq \tau \leq 11$.

[1]For notational facility

TABLE II. CODE OF SOME SELECTED SCALES AND ITS FORTE NUMBER (x MEANS ANY TONIC VALUE BETWEEN 0 AND 11).

Scale	Code	N^o Forte
Chromatic	[0 12 1 11 1 x]	12-1
C Major	[0 7 1 3 2 0]	7-35
A natural minor	[0 7 1 3 7 9]	7-35
Hexatonic	[0 6 1 1 1 x]	6-35
Octatonic	[0 8 1 4 7 x]	8-28

Table II shows the coding system proposed for a few scales. In the last column can be seen as the Forte number is equal for several modes the same scale.

B. Scales encoding algorithm

Since the proposed scales identification method returns the scale code and not your name, then is necessary to use an algorithm that automatically determines the code of a scale from its notes. This task is performed by the encoder algorithm, which is described below.

For a scale of n notes whose tones are stored in the vector $\mathbf{e}_{1\times(n+1)}$, the encoding algorithm returns the code vector $\mathbf{c}_{1\times 6}$. The encoder consists of two parts, the first one validates if the structure scale corresponds to the structure of an analysis scale and calculates the values t_m and n, and the second one calculates the values g, η and m.

The vector $\mathbf{e}_{1\times(n+1)}$ contains all the pitches of the scale in ascending order (in MIDI values) including the octave. The value of the first element of this vector corresponds to the tonic (τ), thus the last code vector value is defined, i. e., $\tau = e_1$. The next step is to calculate the vector of intervals between successive notes, thus: $I_j = |e_{j+1} - e_j|, \quad 1 \leq j \leq (n-1)$. To this vector is determined its structure $u\left\{\hat{s}, \hat{t}, \hat{t}_m\right\}^2$, which must be validated before proceeding.

1) Intervallic structure validation $u\left\{\hat{s}, \hat{t}, \hat{t}_m\right\}$: The first step consists of to test if $\hat{t}_m \in \{0, 1, \ldots, 4\}$. If this condition is fulfilled, then we test whether the value of $\hat{n} = \hat{s} + \hat{t} + \hat{t}_m$ is suitable, i.e., if $n_{\min} \leq \hat{n} \leq n_{\max}$, where $n_{\min}$ and $n_{\max}$ are the limits of n as a function of t_m, which are defined as:

$$n_{\min} = 1/2 \cdot (12 - t_m + (t_m \bmod 2)), \tag{1}$$

$$n_{\max} = 12 - 2t_m. \tag{2}$$

The validation of s and t is defined in similar way, i.e., it is verified if $s_{\min} \leq \hat{s} \leq s_{\max}$ and if $t_{\min} \leq \hat{t} \leq t_{\max}$. The limit values of s, t and n should all be as a function of the known parameter t_m. The equation for $s_{\min} = f(t_m)$ and $s_{\max} = f(t_m)$ is, respectively:

$$s_{\min} = t_m \bmod 2, \tag{3}$$

$$s_{\max} = 12 - 3t_m. \tag{4}$$

The value of $t_{\min}$ is always 0, and $t_{\max} = f(t_m)$ is defined as the half of the difference between s_{max} and s_{min}, so: $t_{\max} = (s_{\max} - s_{\min})/2$. Substituting Eq. (3) and (4) in the previous definition is obtained:

$$t_{\max} = 1/2 \cdot (12 - 3t_m - (t_m \bmod 2)). \tag{5}$$

If the validation $\hat{s}$ and $\hat{t}$ are successful then it is necessary to validate if these values agree with the theoretical values, which are defined as a function of known parameters t_m and n. Using Eqs. (1) and (4) the expression for $s = f(n, t_m)$ and $t = f(n, t_m)$ is, respectively:

$$s = 2n - 12 + t_m, \tag{6}$$

$$t = 12 - 2t_m - n. \tag{7}$$

If the theoretical values (s and t) are equal to validation values ($\hat{s}$ and $\hat{t}$) it is confirmed that this values belonging to one of the scales of analysis and we can continue with the second part.

2) Calculating g, η and m: For calculating the number of the group g we must first calculate a vector $\mathbf{E}$ with the total number of primary scales within each group g and a vector $\mathbf{M}$ with the total number of secondary scales generated by each primary scale. The $\mathbf{M}$ vector is calculated as:

$$M_i = \frac{n}{d_i}, \quad i = 1, 2, \ldots, |\mathbf{d}| \ , \tag{8}$$

where $\mathbf{d}$ is a vector with the divisor of the Greatest Common Divisor (GCD) of the elements of $u\{s, t, t_m\}$, where d_i is the ith divisor and $|\mathbf{d}|$ its cardinality, equal to the total number of divisors. The vector $\mathbf{E}$ is calculated from the following system of two equations with $|\mathbf{d}|$ incognitos[3]:

$$\begin{array}{llll} (1) & PR & = & \sum_{1}^{|\mathbf{d}|} M_i E_i \\ (2) & PCR & = & \sum_{1}^{|\mathbf{d}|} E_i \end{array}, \tag{9}$$

where PR is the total number of secondary scales, equal to the sum of the total number of modes of each primary scale, and PCR the total number of primary scales. The total number of secondary scales is calculated thus:

$$\begin{array}{lll} PR^n_{s,t,t_m} & = & {}_nC_{(n-s)} \cdot {}_{(n-s)}C_t, \\ & = & {}_nC_{(s+t)} \cdot {}_{(s+t)}C_s. \end{array} \tag{10}$$

The total number of primary scales is calculated using the equation for calculating circular permutation with repeated elements (also called necklaces) [9] as follows:

$$PCR^n_{u_j} = \begin{cases} \frac{PR^n_{u_j}}{n}, & |\mathbf{d}| = 1 \\ \frac{1}{n} \sum_{1}^{|\mathbf{d}|} \phi(d_i) \frac{(n/d_i)!}{\prod (u_j/d_i)!}, & \text{otherwise} \end{cases}, \tag{11}$$

where $\phi(\cdot)$ is the Euler's totient function [13] and j is an index used to traverse the elements of the structure $u\{s, t, t_m\}$.

[2]The symbol $\wedge$ indicates that the values have not yet been validated

[3]For structures that use only s, t and t_m the maximum number of incognitos is 3 ($|\mathbf{d}| = 3$)

When the number of divisors is equal to 2 the solution of the System of Equations (9) is given by:

$$E_i = \; (-1)^i \, \frac{d_1 \cdot d_2 \cdot PCR - PR}{d_{(i-(-1)^i)} (d_2 - d_1)}, \quad i = 1, 2. \tag{12}$$

For the case of three divisors the system is solved using the Diofanto's method and we obtain the following inequality for E_1:

$$\frac{PR - M_2 \cdot PCR}{M_1 - M_2} < E_1 < \frac{PR - M_3 \cdot PCR}{M_1 - M_3}. \tag{13}$$

After finding an integer solution for E_1, the system is solved by finding the values for E_2 and E_3, thus:

$$E_2 = \frac{(PR - M_3 \cdot PCR) - (M_1 - M_3)\, E_1}{(M_2 - M_3)}, \tag{14}$$

$$E_3 = \frac{(M_2 \cdot PCR - PR) + E_1 \,(M_1 - M_2)}{(M_2 - M_3)}. \tag{15}$$

The circular permutations with repeated elements (PCR) of n elements and k classes are generated using the algorithm described in [9]. The algorithm is executed for the elements of the structure $u = \{s, t, t_m\}$, and invoked as $\mathbf{H} = \mathbf{N}_k^n\,(s, t, t_m)$, where the total number of elements is $n = s + t + t_m$ and the number of classes is $k = 3$. The vector $\mathbf{p}$ with the period of each primary scale of $\mathbf{H}$ is calculated as follows:

$$p_j = \left\{ a \,\middle|\, \text{if } H_{j,(1:a)} = H_{j,(ia+1:ia+a)} \;\forall i \in [1..b] \right\}, \tag{16}$$

where a varies between M_1 and $M_{|\mathbf{d}|}$, b is equal to $(n-a)/a$ and j is a index between 1 and PCR. Subsequently, the rows of the matrix $\mathbf{H}$ are ordered in ascending order according to the period vector.

Let $\phi(\cdot)$ be a function that returns the exact position of a vector into a matrix by the row index a and the number of circular rotations b, operation defined as $[x, y] = \phi(\mathbf{H}, \mathbf{I})$; and let $\delta(\mathbf{v}, k)$ be a function that returns the element positions of the vector $\mathbf{v}$ equal to k. Finally, the number of the group g is equal to $\delta(\mathbf{M}, \mathbf{p}_x)$, and the number of the scale η within group g is equal to $\delta(\mathbf{w}, x)$, where $\mathbf{w}$ is a vector containing the row numbers of the matrix equal to the period at position x, this is: $\mathbf{w} = \delta(\mathbf{p}, \mathbf{p}_x)$. At last, the mode m is the number of rotations y.

C. Scale detecting algorithm

The algorithm determines cumulatively the interval vector of a set of notes and then calculates its structure. The correct structure of the scale is determined by a deterministic walk conditioned by rules through the nodes of the graph of valid structures, which is shown in Fig. 1. In this graph a node represents each of the 19 possible structures when used intervals S, T or T_m on a scale, and the edges represent the relationship between the valid structures, indicating how one can move from one structure to another dividing the intervals T and T_m at shorter intervals. Afterwards, the tonic and the modes are determined by means of a search for evidences. With the above data is generated the scale within an octave, invoked the encoder algorithm and obtained the scale code. A more detailed description is presented below.

TABLE III. VALUES OF SOME VARIABLES USED BY SCALES ENCODING ALGORITHM.

n	t_m	t	s	g	$\mathbf{E}$	$\mathbf{M}$	PR
6	0	6	0	1	1	1	1
7	0	5	2	1	3	7	21
				1	8	8	
8	0	4	4	2	1	4	70
				3	1	2	
9	0	3	6	1	9	9	84
				2	1	3	
10	0	2	8	1	4	10	45
				2	1	5	
11	0	1	10	1	1	11	11
12	0	0	12	1	1	1	1
6	1	4	1	1	5	6	30
7	1	3	3	1	20	7	140
8	1	2	5	1	21	8	168
9	1	1	7	1	8	9	72
10	1	0	9	1	1	10	10
5	2	3	0	1	2	5	10
6	2	2	2	1	14	6	90
				2	2	3	
7	2	1	4	1	15	7	105
8	2	0	6	1	3	8	28
				2	1	4	
5	3	1	1	1	4	5	20
6	3	0	3	1	3	6	20
				2	1	2	
4	4	0	0	1	1	1	1
Total					**132**	**87**	**927**

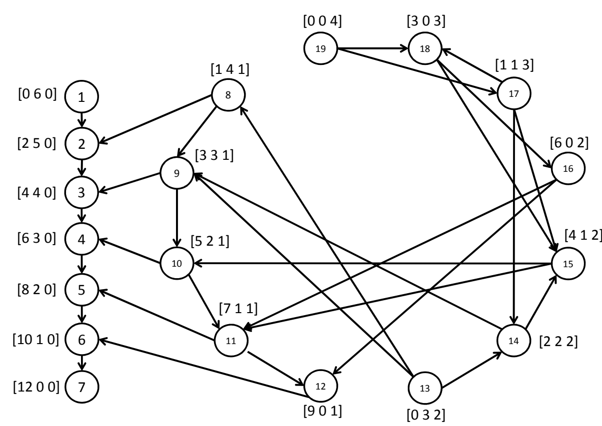

Fig. 1. Graph of the relationship between the structures $u\,\{s, t, t_m\}$.

For a given melody the first j pitches of the vector $\mathbf{z}$ are extracted, thus: $q_i = z_i, 0 < i \leq j$. The vector $\mathbf{q}$ is normalized by means of the operator module and then determined its unique elements, and we obtain $\bar{\mathbf{q}}^{(u)} = \mu\,(\mathbf{q} \bmod 12)$, where $\mu(\cdot)$ is a function that returns the unique elements of a given vector. For this scale is added the octave, so: $\bar{\mathbf{q}}_{l+1}^{(u)} = \bar{\mathbf{q}}_1^{(u)} + 12$, where l is the size vector.

At the initial state, it must be determined what will be the first node of the walk. For this purpose, from the vector $\bar{\mathbf{q}}^{(u)}$

is calculated the interval vector $\mathbf{I}$ and its respective structure $u\left\{\hat{s}, \hat{t}, \hat{t}_m\right\}$. This structure is named current structure u_A and validated according to Section II-B1. If it is valid then the first node of the walk in the graph has been reached, and the structure is stored in a temporary structure u_B, which is initialized with zeros $u_B = \{0, 0, 0\}$.

The next node is determined adding the next pitch and repeating the process. If the new current structure is different to the previous ($u_A \neq u_B$), then the node of this temporary structure is the next node of the walk. The process is repeated until stabilized at an end node or until it reaches a stop condition. The end node corresponds to the structure scale of the melody.

Thereafter, the tonic is determined by analyzing the pitch-class distribution of the melody and performing a search method for evidence [21]. With the interval vector, the tonic and the mode is generated the scale and the scale code $\mathbf{c}$ is obtained using the encoding algorithm.

A special case occurs when the initial node can not be found. This indicates that no structure is valid because they were not used all the notes of the scale in the composition. Thus the interval vector is incomplete and the structure will contain intervals greater than T_m. This case is discussed below.

1) Scales with incomplete structure: Note that, for this case, at the end of the process the structure u_A corresponds to the structure of all intervals of melody, including intervals with more than 3 semitones. Possible interval structures with intervals of S, T and T_m that can be formed with u_A are determined through a method based on partition of an integer number. A matrix $\mathbf{W}$ that contains the partitions of the total number of semitones which form intervals greater than T_m is calculated so: $\mathbf{W} = \theta\left(s_t, \left[\begin{array}{ccc} 1 & 2 & 3 \end{array}\right]\right)$, where $\theta(\cdot)$ is a function that returns the partitions of an integer for some given candidates, in this case $\left[1 \;\; 2 \;\; 3\right]$, and s_t is the sum total of semitones forming intervals greater than T_m. The array elements with the structures that can be formed with u_A are calculated thus: $u_i = \;\; w_i + u_c, \quad 0 < i \leq l$, where u_c is the the structure formed by the first three values of u_A and l is the total partitions (possible structures). For all modes of every possible structure is calculated the Mean Percentage of Harmonic Similarity (MPHS), which measures the similarity between the intervals of the scale and the spectral characteristics of the set of intervals of the harmonic series [12]. The scales (or modes) commonly used are those containing high MPHS value. The major diatonic scale has MPHS equal to 39.61.

III. Experimental Results

In this section, the proposed method is applied to detect the musical scale of a monophonic melody without modulation. Specifically, the goal is to identify the scale code of melodies composed using both traditional and rare scales.

The first database tested was Finnish folk songs, which contains 8613 short melodies in major or minor scale [19]. However, only 6691 instances have tonality information within the database. Figure 2 shows the structure evolution of melody No. 28, where it is possible see how the melody begins with an invalid structure until the 5th pitch, and from the 6th pitch

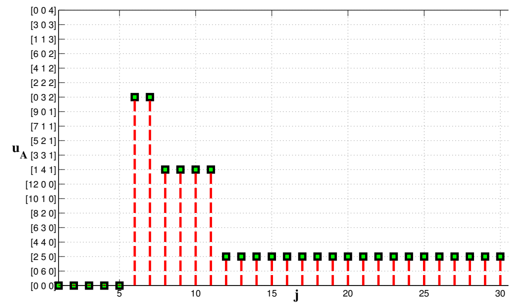

Fig. 2. Evolution of structures inside the graph for the melody No. 28 of the Finnish Folk Tunes database.

TABLE IV. Total number of possible structures, percentage of melodies and accuracy.

l	Total	Percentage (%)	No. correct	(%) Accuracy
1	1161	34.57%	0	0.0%
4	783	11.70%	311	39.72%
5	796	11.90%	326	40.96%
7	40	0.60%	10	25.00%
8	388	5.80%	123	31.70%
10	109	1.63%	64	58.72%
12	75	1.12%	37	49.33%
14	2	0.03%	0	0.00%
16	4	0.06%	0	0.00%
Total	6691	100%	3055	45.68%

it passes through two nodes (structures $\{0, 3, 2\}$ e $\{1, 4, 1\}$) before stabilizing at the node corresponding to the structure of the diatonic scale ($\{2, 5, 0\}$). The encoding algorithm is executed for the latter structure and scale code is calculated. The scale code calculated for this melody is [0 7 1 3 2 0], which belongs to the C major scale, confirming the key indicated in the database.

Considering only the melodies that use all notes of the scale, which correspond with 52.25% (3496 melodies), the algorithm correctly detect the scale of 3303 melodies, obtaining an accuracy of 94.48%. For melodies using incomplete scale, in Table IV is shown the total number of melodies by total possible structures (l) and its respective percentage in the database.

In the fourth column is shown the amount of melodies whose scale has been correctly detected and the fifth one the corresponding accuracy. For example within the database there are 783 melodies with four possible structures, which corresponds with the 11.70% of the total, and 311 were correctly identified, obtaining an accuracy 39.72%. The error is mainly due to the tonic identification τ, since sometimes there is not enough evidences for its identification. On the other hand, note that for $l = 1$ the accuracy is 0% because although the scale is incomplete its structure falls in any of the 19 valid structure analysis.

For melodies with incomplete scale, the algorithm estimates what are the possible structures and selects the which has the greater MPHS. For example, the tone returned by Krumhansl-Schmukler algorithm for the melody No. 2 was F major, but the pitch-distribution shown in Fig. 3 reveals that only 6 of 7 notes of the scale were used, exactly the D note was omitted. Although Krumhansl-Schmukler algorithm has detected the F major key, the lack of this note creates confusion in the tonality. The proposed algorithm shows that this

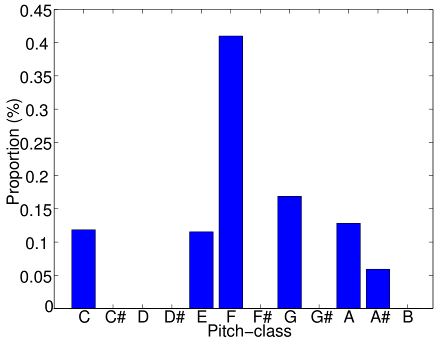

Fig. 3. Pitch-class distribution of the melody No. 2.

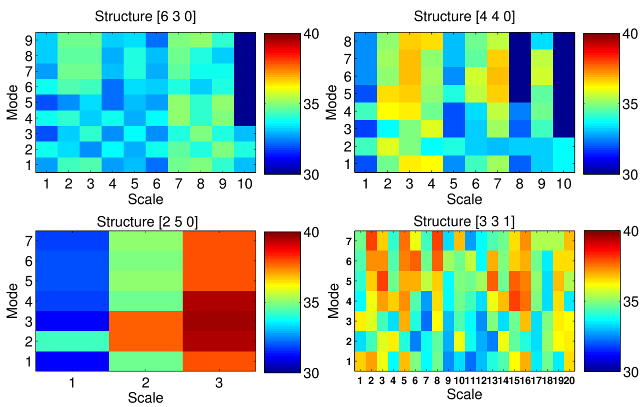

Fig. 4. MPHS value of all scales of the possible structures of the melody No. 2.

melody has 4 possible structures $\{6, 3, 0\}$,$\{4, 4, 0\}$,$\{2, 5, 0\}$ and $\{3, 3, 1\}$. Using Eq. (11) the number of primary scales that can be created with these structures is calculated, which is equal to 10, 10, 3 e 20, respectively, for a total of 43 primary scales. Figure 4 shows the colormap matrix with MPHS values of the 4 possible structures. The largest value, equal to 0.79, corresponds to the structure $\{2, 5, 0\}$, which is the structure of the major/minor scale. The scale code returned by the detector is [0 7 1 3 2 5], which is the code of the F major scale.

The second database contains random melodies composed using one of 927 secondary scales and using the full scale. The number of notes melody was set at 100, the number of octaves at 2 and the tonic was randomly selected. The algorithm correctly detected the scale code of 894 melodies, giving an accuracy of 96.4%.

IV. Conclusions

In this paper we propose a method for identifying the scale of a melody in a comprehensive manner. This method can detect both known and unknown scales. To standardize and facilitate the handling and naming of scales, a coding system was introduced as well as the respective encoder algorithm. For the Finnish melodies database, the algorithm got high accuracy when using melodies with full scale. Therefore, for scales with incomplete structure the algorithm made a good estimate of the scale by measuring the percentage of harmonic similarity. A special case took place to incomplete scales but with valid structure, where the algorithm is able to correctly identifies the scale used but in the database it is labelled with major or minor scale. What does it mean to use an incomplete scale can turn the scale in another and generate ambiguity. For the database of random melodies composed of both known and unknown scales the method showed high accuracy. However, the main difficulty is to identify the mode and tonic, due to the fact that melodies do not follow a method of modal or tonal composition. Therefore, to correctly identify a musical scale, it is also necessary to consider the type of composition in the original melody. As a future work, we will propose to extend the method to scale with intervals greater than 3 in its structure and also to scales with tempered microtonal tuning system (N-TET with $N > 12$). It also aims to improve the estimation method of incomplete scales and adapt it to operate with non-monophonic melodies and with modulation.

References

[1] Coca A., Tost G., and Zhao L., *Characterizing chaotic melodies in automatic music composition*, CHAOS **20** (2010), no. 033125, 1–12.

[2] Coca A. and Zhao L., *Musical rhythmic pattern extraction using relevance of communities in networks*, Information Sciences **329** (2016), 819–848.

[3] Forte A., *The structure of atonal music*, Yale University Press, 1977.

[4] Krumhansl C., *Cognitive foundations of musical pitch*, Oxford University Press, 1990.

[5] Krumhansl C. and Kessler E., *Tracing the dynamic changes in perceived tonal organization in a spatial representation of musical keys*, Psychological Review **89** (1982), no. 4, 334–368.

[6] Lucy C., *Pitch, PI, and other musical paradoxes, (a practical guide to natural microtonality)*, LucyScaleDevelopments, 1994.

[7] Parsons D., *The directory of tunes and musical themes*, S. Brown, 1975.

[8] Temperley D., *What's key for key- the Krumhansl-Schmuckler key-finding algorithm reconsidered music perception*, Psychological Review **17** (1999), no. 1, 65–100.

[9] Ruskey F. and Sawada J., *An efficient algorithm for generating necklaces with fixed density*, SIAM J. Computing **29** (1999), no. 2, 671–684.

[10] Fripertinger H., *Enumeration and construction in music theory*, In Proc. Diderot Forum on Mathematics and Music, 1999, pp. 179–204.

[11] Attakitmongcol K., Chinvetkitvanit R., and Sujitjorn S., *Characterization of traditional Thai musical scale*, In Proc. WSEAS Trans. on Acoustics and Music **1** (2004), 90–95.

[12] Gill K. and Purves D., *A biological rationale for musical scales*, Plos One **4** (2009), no. 12, e8144.

[13] Goldberg K., Newman M., and Haynsworth E., *Handbook of mathematical functions with formulas, graphs, and mathematical tables*, ch. The Euler Totient Function, p. 826, Dover publications, 1972.

[14] Longuet-Higgins and Steedman M., *On interpreting Bach*, Machine Intelligence **6** (1971), no. 221, 221–241.

[15] Keith M., *Combinatorial music theory: Adventures in musical combinatorics*, Vinculum Pr., 1991.

[16] Levine N., *Exploring algorithmic musical key recognition*, Claremont McKenna College - CMC, Senior Theses, 2015.

[17] Slonimsky N., *Thesaurus of scales and melodic patterns*, Amsco Publications, 1947.

[18] Dutta S., Kumar C., and Chakraborty S., *A symmetric key algorithm for cryptography using music*, International Journal of Engineering and Technology **5** (2013), no. 3, 3109–3115.

[19] Eerola T. and Toiviainen P., *Suomen Kansan eSävelmät: Digital archive of finnish folk songs [computer database]*, University of Jyväskylä, 2004.

[20] Collier W. and Hubbard T., *Musical scales and evaluations of happiness and awkwardness: Effects of pitch, direction, and sacle mode*, American Journal of psychology **114** (1994), no. 3, 335–375.

[21] Yongwei Z., *A symbolic key finding algorithm*, MIREX, 2005.